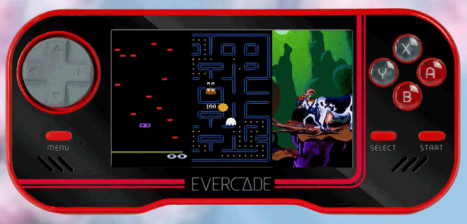
World RetroMagazine
future days are back
ISSN 3103-697X
Issue 7 Year 2 - May 2021 - https://www.retromagazine.net - Free Publishing

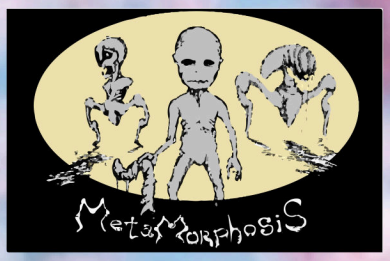


EVERCADE
The retro console
of Blaze Entertainment

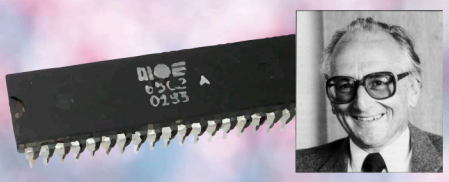
PC Engine

The small giant from the Land of the Rising Sun!

BASIC 10 - LINER CONTEST 2021

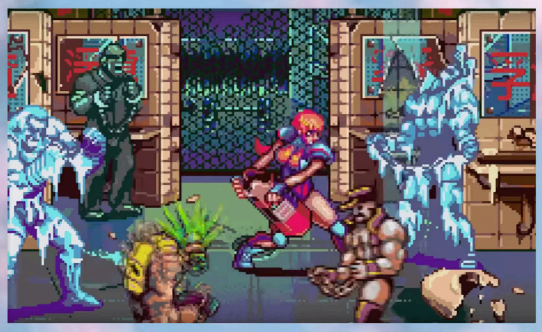


NEW GAME:
METAMORPHOSIS (ZX Spectrum)



CPU: MOS 6502

Emulator's test:
DOSBox ver. 0.74-3



NEW GAME: PAPRIUM (Megadrive)

The Last Ninja (C64 - C128)

The I.C.L. One-Per-Desk computer

Inside:
- KICK OFF 2 - when the football is protagonist!
- Wonder Boy in Monster World
- An introduction to Hollywood - Part 4
- Simulation of PLOT X,Y in the C64 Bitmap mode
- SpeedCalc 2019 - Excel for the C64? Well, maybe...

"Seeing" retrogames through their sounds

Everyone tells me that I always try to "see" the beauty of things and I turn the difficulties I encounter into my strengths.
And now I'm here, 40 years old, playing, having become blind, with my head full of 8-bit memories. For once, a strength of mine that doesn't come out of a difficulty but out of a lot of good memories.

I have always been tied to the sounds of those times, those sounds that today are the only thing that reminds me the emotions of my teenage years, when the only thought was to come home from school to sit down in front of my MSX and later my Amiga 500 Plus and try out new games and software.

How nice to remember the sound of the MSX cassette recorder or Amiga's floppy drive head.

And what about this pandemic period, the first months of forced lockdown, when I discovered this great magazine? With it I was able to turn my passion for the world of retro-computing back on!

So I asked myself: where did I put this great passion of mine for more than 20 years? When did I forget about it? Perhaps, as it happened to many of us, I did leave it in my basement/attic or I had lost it in one of the many moves.

What I miss the most now is the images coming out of the cathode tube screens, all those knotted cables, the yellow color of Amiga's case, the black joysticks with the red buttons and the small sprites of Sensible Soccer players. But thanks to RetroMagazine World, the many evocative videos on YouTube and the many Facebook pages dedicated to this magical world, I too can relive those fairy moments and see again those colourful screenshots in the form of sounds.

So I started exploring other worlds, reading, documenting myself. I have discovered so much in such a little time, but much more I have to discover, learn, read and listen to.

How many machines I didn't know about, how many games and software, but even more, how many fans!

May this wonderful world of retrocomputing and retrogaming never end and continue to cheer us up, give us so many emotions, with an eye (or an ear) always looking forward to the future.

Alessandro Albano

CONTENTS

◊ The I.C.L. One-Per-Desk computer	Pag. 3
◊ Evercade - Blaze Entertainment	Pag. 4
◊ PC Engine - The small giant from the Land of Rising Sun	Pag. 6
◊ The MOS 6502 CPU	Pag. 8
◊ Structuring old BASIC dialects with For-Next loops	Pag. 12
◊ BASIC in a nutshell: waves on LM80C and MSX-1	Pag. 14
◊ Grapheur 1.0 - Doing graphs with the Amstrad CPC	Pag. 16
◊ SpeedCalc 2019 - Like having Excel on a C64? Well, almost...	Pag. 18
◊ Simulating PLOT X,Y in C64 bitmap mode	Pag. 20
◊ May the FORTH be with us - part 3	Pag. 22
◊ Basic 10-Liner Contest 2021	Pag. 23
◊ Introduction to Hollywood - part 4	Pag. 26
◊ Emulators: DOSBox ver. 0.74-3	Pag. 30
◊ Kick Off 2, when the football is the protagonist	Pag. 32
◊ The Last Ninja: legends never die	Pag. 35
◊ Paprium (MD)	Pag. 38
◊ Runn 'n' Gunn (C64)	Pag. 40
◊ Wonder Boy in Monster Land (MD)	Pag. 41
◊ Smarty and the Nasty Gluttons (Amiga)	Pag. 42
◊ Chrono Trigger (SNES)	Pag. 44
◊ Metamorphosis (ZX Spectrum)	Pag. 46
◊ Gravibots (MegaDrive)	Pag. 48
◊ Monstro Giganto (C64)	Pag. 49
◊ Neptune Lander Elite (C64)	Pag. 50
◊ Inviyya (Amiga)	Pag. 52
◊ Xain'd Sleena (Arcade)	Pag. 53
◊ Pub Trivia Simulator (C64)	Pag. 54

People involved in the preparation of this issue of RetroMagazine World (in no particular order):

- Alberto Apostolo
- Gianluca Girelli
- Antonino Porcino
- Carlo N. Del Mar Pirazzini
- Daniele Brahimi
- Mic the Biker Novarina
- Francesco Fiorentini
- Takahiro Yoshioka
- Attilio Capuozzo
- Francesco Bizzini
- Leonardo Miliani
- Querino Ialongo
- Roberto Del Mar Pirazzini
- Cover & graphics by Flavio Soldani
- Revisioning & proofreading by DLM

A bit of rarity

(rummaging here and there)

The I.C.L. One-Per-Desk computer

by Alberto Apostolo

When it appeared on the market in 1984, the One Per Desk computer (Fig.1) represented an innovative hybrid between a Personal Computer and a tele-communications terminal.

Built by International Computer Limited, it was based on Sinclair QL hardware and was the result of a collaboration between Sinclair Research and British Telecom that began in 1983.

The same model was known as Merlin M1800 Tonto (The Outstanding New Telecoms Opportunity) and also as Computerphone (at Telecom Australia and New Zealand Post Office).

Like the Sinclair QL, it used a Motorola 68008 CPU, ZX8301/ZX8302 ULA chip, 128KB RAM.

Unlike Sinclair QL, an Intel 8051, two PSTN (Public Switched Telephone Network) lines, a V.21/V.23 modem, a TI TMS5220 speech synthesizer (used for answering machines as in answering machines) were mounted on the O.P.D. computer.

The available I/O drives were a 9 inch monochrome monitor (hereinafter a color monitor) and two 3.5 inch disk drives.

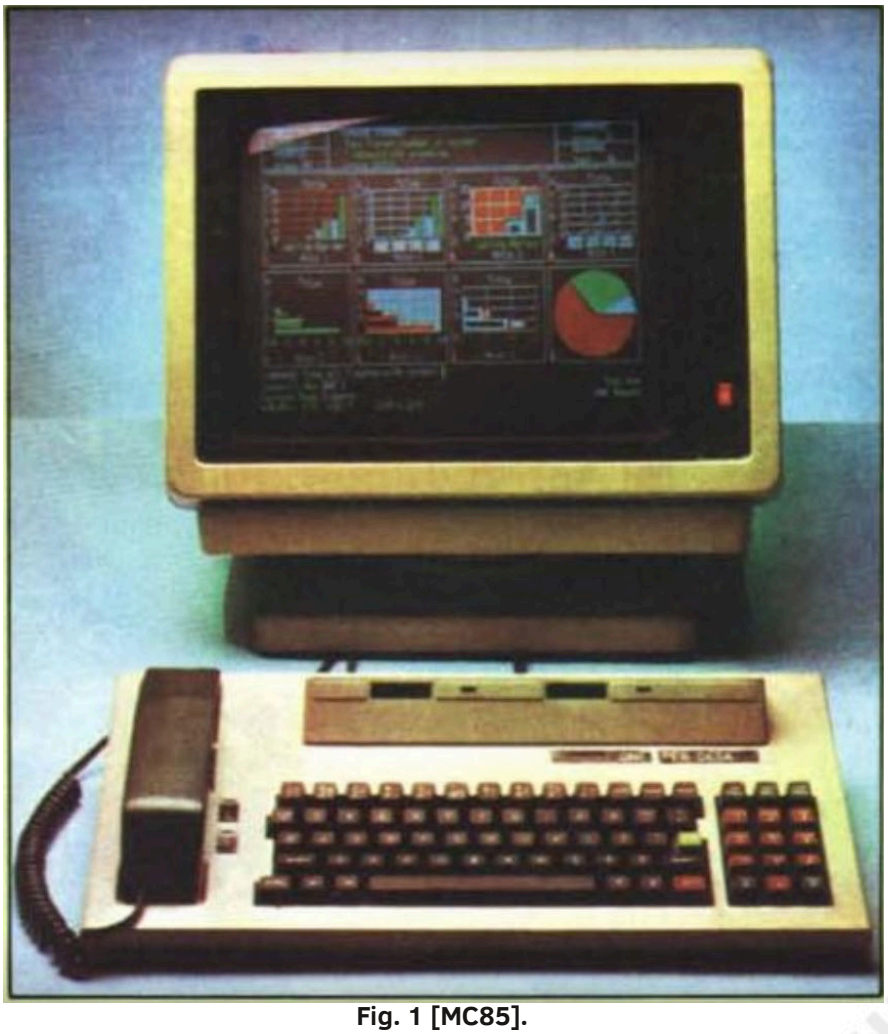


Fig. 1 [MC85].

The Basic Functional Software (BFS) operating system had no similarities to the QDos despite a subset of the SuperBASIC Sinclair being provided.

XChange was a suite available for O.P.D., wired in ROM and for sale for £130. Made by Psion, it was similar to the suite of QUILL, ABACUS, ARCHIVE and EASEL programs that ran on the Sinclair QL.

Initial orders amounted to £4.5 million (1500 copies) from British Telecom and £8 million from Telecom Australia.

Bibliography

[Wi21] retrieved in 2021/03/29 from https://en.wikipedia.org/wiki/One_Per_Desk

[MC85] MC Microcomputer n.42 Jun 1985, pag.16, https://issuu.com/adpware/docs/mc042

EVERCADE - Blaze Entertainment

by Carlo Nithaiah Del Mar Pirazzini

The new portable consoles that wink at retrogaming. This is the “new trend” that is changing the general picture of emulation... And beyond.
Today we present Evercade of Blaze Entertainment.
This cute portable console wants to be a new interpretation of the "walking” and old-school game.
A new way to reinterpret the game or something that should stay on store shelves?

Released at the end of 2020, Evercade jumps head-on abnormally despite other well-known retrogaming devices. It does it in the most classic way possible... the cartridges! Gentlemen, the classic system of good times!!!
A selection of games chosen instead of billions of Roms on an SD card. It avoids paralysis of choice and I have to admit that the producers have done things correctly, with a license for each game (Everything is Super Regular).
Let's take a closer look.

THE STRUCTURE

Evercade does not allow space for further storage on board. Games arrive on real cartridges that fit on the console. An anomaly in a world of streaming, downloading and other devilishness.
Each cartridge comes with a color instruction manual that outlines the history and rules for each game. Nice!!! Really nice!
A widescreen display on a handheld that mainly plays

games in 4:3 ratios may seem odd; but Atari Lynx games require 16:9 for example and black sidebars for other systems are tolerated.
I would have preferred a better quality video, but overall it's fine.
The big D-pad works fine, while the buttons and triggers on the backs are really nice but I found them “fingertip slaughter.”
In fact, L’Evercade as a whole “feels good” in his hand.
It's heavy enough to look SUBSTANTIAL, but comfortable for long sessions.

But how does playback work? With emulation logically. The console has not magically managed to hide the tide of platforms inside it and so it turns directly to emulators.

Fortunately, the emulation is good. No perceptible delay, the framerate is solid and everything turns more than well. You can also use the save state for progress in many games.

GAMES

Different cartridges are available and you are well over 100 games.
The complete selection can be viewed on the console

website. Among the collections I cite some.

There are both classic and new title collections. Very positive. **Mega Cat Studios Collection 1**: The strongest selection - is this set of modern titles for vintage systems. Each stock is sold for a small fortune in its original form, so ten of them for 17 euros is a real bargain. Tanzer, Old Towers and Super Painter are amazing.

Atari Lynx Collection 2: Both Lynx collections are quality, but this second set looks good with After Burner's aspiring Blue Lightning, the crucial and cruel Chips Challenge puzzle and California Games multi-challenge sports simulator (which Atari Lynx version is a bomb).

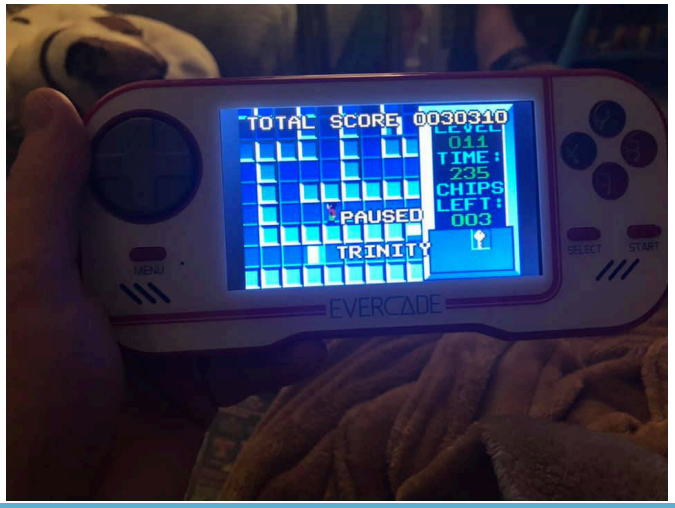
Namco Collection 1: I would have been happier with arcade ports, but the NES versions of Pac-Man, Dig Dug and Xevious work pretty well here. In addition, there are eight other titles, including Battle Cars SNES racing.

Piko Interactive Collection 1: There are 20 new / old titles in this Piko set, collecting the incomplete game pieces and finishing them. Highlights include the Switchblade action platform and the fast Top Racer.

Atari Collection 1: You would expect Atari VCS (Adventure) successes, but it also includes fantastic Atari 7800 curiosities like Food Fight and extravagant sports/ Karateka mash-up Ninja Golf. I particularly love the 7800 and I really liked this as a set.

VERDICT

It is true that Evercade has irregular edges. The screen is simply OK but nothing more and you could have made



better decisions about game formats. In some cases I would have preferred to see porting directly from Arcade than conversions from NES.

But there is no denying that this is an idea made by people who really care about the history of games. In addition, Evercade has not stopped with its first ten cartridges: for example, some very special ones have emerged: that of Codemasters (including Sensible Soccer), a collection of Worms and the 14 Indian games. According to reports, the original arcades will also be arriving soon.
Given the affordable price, “old-school" feel and sense of concentration, this strange laptop is an investment worth making.

You can find all the information for the console at:
https://www.evercade.co.uk/

In Italy instead you can take a look at Coinup Italia (which I thank for the test)

TECHNICAL SPECIFICATIONS

Screen: 4.3 inches 480 x 272
Processor: 1.2 GHz Quad-core
RAM: 256 MB
Connectivity: MiniHDMI output (720p)
Battery life: Up to 5 hours
Size: 182x85x23mm, 216g

FINAL SCORE

PROs

Collectible Cartridges
Reactive controls
Good selection of games
Save state for each title
Affordable price and many possibilities to customize your Bundle

CONs

Display not really top of the range.

PC ENGINE – The small/giant from the Rising Sun!

by Takahiro Yoshioka and Carlo N. Del Mar Pirazzini

Originally released by the Japanese consumer electronics giant NEC in 1987, the PC Engine was the cartridge-based system for the most avid players.
Designed to compete with the Nintendo Famicom, the machine was an innovative joint venture with veteran developer Hudson Soft, which boasted a 16-bit graphics processor and probably the best audio chip of the time.
Released much later in the US as TurboGrafx-16, and later again in the UK under the captivating name PC Engine CoreGrafx, it never had the momentum to challenge Sega or Nintendo, but its production of faithful arcade conversions and brilliant sliding shooters ensured its cult charm.

But let's take a look at its story.
Hudson Soft had a project: to create a new console that did not use cartridges as a game storage medium. He proposed the idea to NINTENDO with whom he had an excellent relationship. Osaka's house refused, but NEC seized the opportunity to enter the console market using the know-how of Hudson Soft's technicians.

It was released on October 30, 1987 and was considered one of the most compact consoles ever made. The name PC Engine, in addition to recalling the link with the series of personal computers produced by Nec, was chosen to indicate the expansion capacity of the system.
In April 1988 the PC Engine reached 1.5 million units sold, conquering the Japanese video game market. In October of the same year, the console competed with the new Sega Megadrive.
In the summer of 1989, NEC marketed the console on the US market under the name TurboGrafx 1. It was an immediate success.

The American version was aesthetically different from the larger, black PC Engine, but used the same type of cartridges as its Japanese counterpart. The console was also able to play games on CD-ROM (which was innovative at the time) thanks to the TurboGrafx CD, the American version of the external CD player CD-ROM for PC Engine. The TurboGrafx was also released in Europe, in the UK

market, and with the precaution to make the console compatible with the PAL standard.
Three different versions of the PC Engine were released in Japan in 1989: Shuttle, CoreGrafx and SuperGrafx. The latter models were distributed in France by Sodipeng, while the Shuttle (inexpensive model not compatible with the CD ROM) was unsuccessful.
The SuperGrafx had to compete directly with the Nintendo console that would replace the Famicom/Nes, the Super Famicom. However, in addition to being compatible with the PC Engine library, it was able to run only five exclusive games.
At the Consumer Electronics Show in 1990 Nec presented the TurboExpress, an incredible portable console compatible with the games of the TurboGrafx 16, enjoying some success.
By 1991, however, the market was already dominated by Nintendo and Sega. Nec tried to reduce the price of the home console and even announced the launch of the TurboDuo, a version of the TurboGrafx 16 with the built-in CD player.
In 92 NEC and Hudson founded Turbo Technologies Incorporated (Tti) which was to deal with the distribution on the American market of the new platform from October of the same year.
Tti tried to reduce the price of the TurboGrafx 16, Turbo Express and TurboGrafx CD even further, but without success.
In 1993 NEC began working on the PC-FX, the next console, but it turned out to be a colossal fiasco.
The PC Engine finally left the scene in 1995.
But entering the legend!

BEST FOR PC ENGINE
Here's a selection of the top five titles:

Dracula X - Castlevania Rondo of Blood
Released on CD and with a truly incredible graphics and sound compartment, this Symphony of the Night prequel

is considered one of the best Castelvania ever. Beautiful!

Snatcher
Very pulp Cyberpunk graphic adventure. A little masterpiece by Hideo Kojima.

Lord of Thunder
Graphically impressive and with compelling gameplay. One of the best shooters for the PC Engine and one of the best ever.

AirZonk
Crazy shooter fast and hilarious.

PC KID
The whole Hudson Soft mascot saga. A particular and very fun platform, a must-see saga on PC Engine.

Final considerations
A large console with an impressive playground ranging from J-RPG to hardcore, from platforms to shoot em up. In 2020, the PC Engine Mini was released, a well-made reproduction containing a number of games to be tested. Your purchase is highly recommended. You can find it on Amazon.

Technical specifications
Custom 8bit CPU: HuC6280 (7.16MHz)
Video Processor: HuC6270
Color Processor: HuC6260
Color Palette: 512
On-Screen Colors: 512 (256 for sprites, 256 for backgrounds)
Resolution 256×216
Sound: 6 stereo channels (5-bit sampling)
Handling of 64 sprites at a time (16 colors and 32×64 maximum size).
Ram: 8 Kbytes (TurboDuo = 32 Kbytes)
Video Ram: 64Kbyte 512 Kbit
Cartridge Size: 256Kbit - 20 Megabit Max. (Street Fighter 2)
Normally 8 Mbits
CD reading speed for Turbo CD / TurboDuo : 1x

PC KID

Dracula X – Castlevania Rondo of Blood

Snatcher

Lord of Thunder

Airzonk

The MOS 6502 CPU

di Leonardo Miliani

We continue our journey through the integrated circuits that have left their mark in IT history. In this episode our focus will be on MOS Technology's 6502 microprocessor, one of the most popular 8-bit CPUs that has been used in a myriad of home computers, consoles and arcade video games between the late 1970s and the first half of the 1980s.

Origins

In the early 1970s Motorola was the second largest semiconductor manufacturer after Texas Instruments. From the production of transistors, which began in the second half of the 1950s, the company had then moved towards the production of computer memories. In the early 1970s, the development of a microprocessor on a single silicon chip began and was introduced in 1974 as the MC6800. Among the developers was Chuck Peddle, who was also in charge of commercial promotion: talking to various potential customers Peddle realized that the market wanted a much cheaper CPU than Motorola's 6800 (it then costed more than $300).

Speaking to his superiors Peddle therefore proposed a simplified and cheaper processor but the project was not accepted. Deeply disappointed by this internel obstructionism and determined to carry out his project, Peddle, after a short time, left Motorola together with other engineers who had developed the processor in search of someone willing to invest in their idea. They then agreed with MOS Technology, a company founded by Allen-Bradley initially to produce chips on behalf of Texas Instruments. The latter, however, in the early 1970s gave life to the so-called "calculator war", starting to sell the assembled calculators on its own and no longer the only chips to build them.

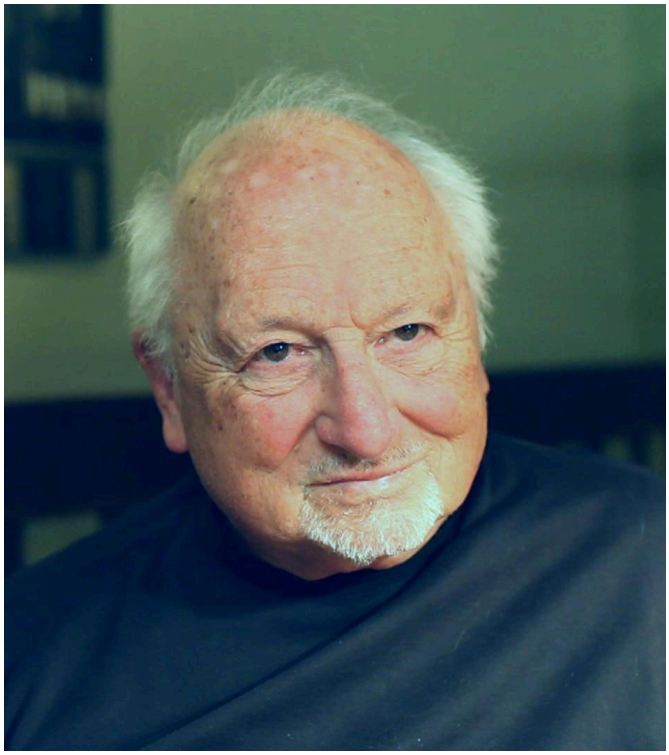


Fig.2 - Chuck Peddle (photo by Jason Scott)

As a result, production collapsed and MOS found itself in trouble and failed looking for other customers, including Atari, which started the production of a version of the Pong on a single chip. Peddle, along with Bill Mensch, Rod Orgill and others, were hired by MOS where they were able to complete the production of the processor they designed. The development of the project led to the creation of a family of CPUs, whose forefathers, presented in 1975, were MOS 6501 and MOS 6502. Compared to the 6800, they had fewer registers and fewer instructions but, due to the reduced complexity and fewer internal transistors, even a much lower production cost.

Fig.1 - MOS 6502 (photo by Dirk Oppelt)

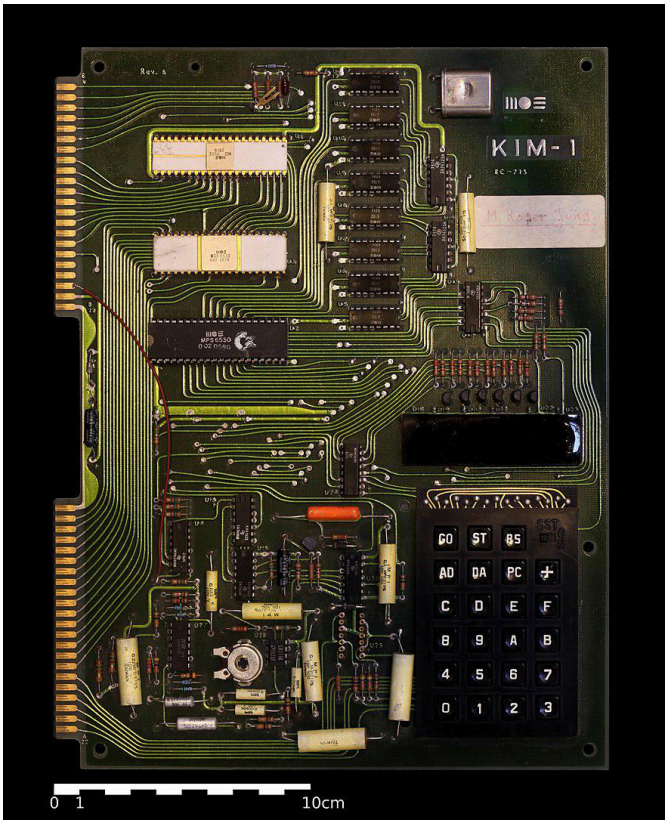


Fig.3 - MOS KIM-1 (photo by Rama and Musée Bono)

These factors, combined with an innovative production process developed by MOS that allowed to limit defects in silicon wafers, thus obtaining many more functioning chips for each production, allowed to market the 2 CPUs at 20 and 25 dollars respectively.

Presented at the Wescom that same year, they attracted a lot of interest, not only from potential customers but also from Motorola... who sued MOS Technology in court believing that its former engineers had designed the processors using knowledge and technologies developed in Motorola labs.

The 6501, moreover, was compatible in terms of pinout with the 6800, thus being interchangeable with it. Allen-Bradley, considering that MOS Technology's business was not going very well, as soon as the subpoena arrived, decided to sell the company, which was taken over by his own employees: Motorola had the money and time to pursue a lawsuit that could bleed the company dry, so MOS' new property settled with Motorola without going on court. The agreements envisaged removing 6501 from the market and paying royalties in order to use the 6800 peripheral chips.

Not only did the company save itself from bankruptcy, but it also started to earn a lot because the 6502 was a huge success, being sold in thousands of units thanks to its low price, a factor for the adoption of this CPU by many computer manufacturers that was much more decisive than the limited characteristics of the chip.

To facilitate the adoption of the CPU, an on-board microcomputer called KIM-1 was created to allow developers to create software for the computers that would adopt the 6502. The KIM-1 was intended for professionals but its relatively low cost and ease of use made it a computer widely used by hobbyists: it was enough to connect it to a terminal (a system composed of keyboard, screen and accessory circuitry that dialogues with a remote processing unit) that obtained a complete computer. Equipping it with a video chip and integrating it into a case with a monitor and keyboard was the logical evolution of that system that led to the commercialization of the Commodore PET.

Architecture

The MOS 6502 is an 8-bit processor with a 16-bit address bus, which can address up to 2^16=65,536 bytes (64 KB) of memory. Its instruction set includes only 56 but there are 3 different addressing modes to compensate for. In order to contain costs, the development group revised the 6800 project and made a number of changes. The 6502 has only 3 internal registers, all with 8 bits: a main register on which most calculations are performed, called A, or “Accumulator", and two index registers, called X and Y, used for addressing instructions. A pointer to the stack “S” and a status register “P” are added to these registers. These registers are also 8-bit. The only 16-bit register is the "PC”, or “Program Counter”, which points the instruction to be executed in memory.

These constructive choices were dictated by some factors: the number of transistors contained in a processor greatly influenced its final production cost, and keeping the number reduced using the minimum of registers meant obtaining a low final cost. In addition, RAMs were, at the time, fairly fast, so it was decided to sacrifice the internal registers and adopt particularly fast instructions in accessing the memory cells, so as to use them as some sort of external registers. The 6502 is in fact equipped with a addressing mode called “zero page” where access

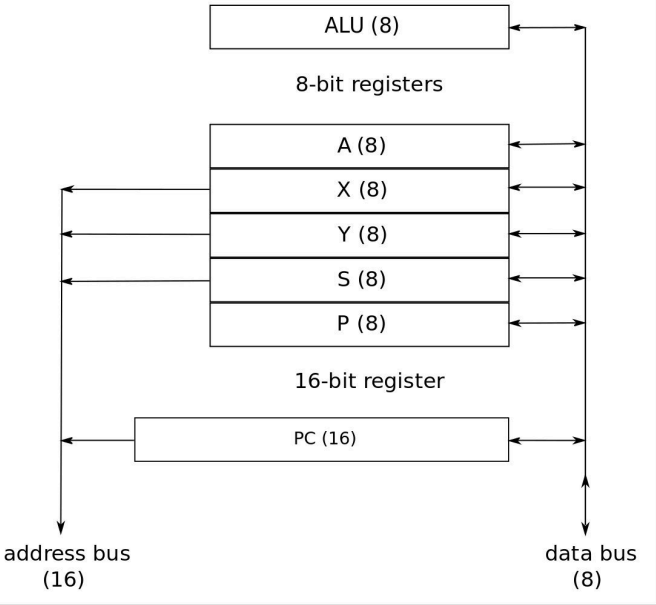


Fig.4 - MOS 6502, inside architecture (from Wikimedia Commons)

to the first 256 bytes of the memory is made using instructions consisting of only 2 bytes, one for the op-code of the instruction and the other for the address: omitting the second byte normally used to compose a 16-bit address avoids further read access to the memory. The stack also follows this philosophy. Since it is mapped in memory on “page 1” to addresses $0100~$01FF, that is, an address space of only 256 bytes, it also needs only 8 bits for addressing. Thanks to these choices, the number of final transistors is only 4,000 (by comparison, the Z80 has more than twice as many).

However, there are instructions for the addition and subtraction in BCD encoding. Binary Coded Decimal (BCD) encoding is a number management system based on the use of 4 bits to represent a decimal digit: going from 0 to 9, you need 4 bits per digit (since 2^4=16). BCD encoding was once used in TTL logic systems without a microprocessor, where simple 7-segment displays were used to display data, driven precisely by 4-line input decoders. The 6502, having also been created as a solution for embedded systems, and the choice to support BCD logic was dictated by the fact that it could thus also be used in such systems where the data could be shown on these displays with little calculation effort.

Again with a view to cost containment, an internal clock generator has been integrated that requires a single phase input signal. The CPU can also handle 2 interrupt levels, one maskable (IRQ) and one non-maskable (NMI). The first one can be deactivated via software while the second one, as the name implies, can't. There are 13 routing modes among which, as mentioned above, there is a quick access mode to “page 0” which makes up for the lack of internal registers. The instructions are 56: for this reason the 6502 is often described as a “Restricted Instruction Set Computer” (RISC), which is a system with a limited number of instructions, although in reality a true RISC processor replaces the limited number of instructions by offering a large number of internal registers, which the 6502 does not have. In addition, RISC systems are also based on a couple of other assumptions: instructions with a fixed length and “load/store” philosophy, that is, to divide the instructions for loading and saving data in memory from those that follow operations in ALU. In these aspects the 6502 appears more like a CISC processor, having both variable length instructions and some mathematical instructions (type “INC”) that also modify the memory.

The maximum clock frequency allowed by the original 6502 design is 1 MHz. Despite this apparent slowness, the 6502 does not disappoint other CPUs, since its designers optimized it to access memory quickly: the 6502 is in fact up to 4 times faster than the processor that inspired it, the 6800. It also doesn't disappoint other CPUs with higher clocks.

The rivalry with the Z80 is famous, if only for the fact that

Fig.5 - An Apple II+. Apple II models are equipped with the 6502 (photo by Bilby)

the latter CPU also appeared in the same period and was used for the same purposes, namely computers and game consoles: the Z80 has a richer set of instructions, many internal registers, a higher working frequency but, on the other hand, requires more clock cycles per instruction, a field in which the 6502 stands out where many instructions are executed in a couple of cycles. Obviously, just analyzing these factors to evaluate the efficiency of a CPU is reductive because both processors excel in certain fields and defect in others but at the time they came out there was a tendency to compare easily comparable elements, and the clock value was the argument used by the supporters of the Z80 while the speed of execution of the instructions was that used by fans of the 6502.

Uses and derivative processors

Due to its low price 6502 was chosen by a large number of hardware manufacturers for the most diverse products. One of the best known isdefinitely the use in Apple computers. Steve Wozniak, in fact, made his Apple I thanks to the fact that the 6502 was put on the market at such a low price. His idea had remained on paper until then because the Intel 8080 and Motorola 6800 cost far more than a student like Wozniak could afford at the time. The 6502 was so important to Apple that the 1977 Apple II,

Fig.5 - PET 2001, first Commodore's computer equipped with the 6502 (photo by Max Mustermann)

sold in its various evolutions until 1993, continued to use this CPU throughout its 16 years of commercialization.

Atari used the 6502 in his wide range of 8-bit computers. The first model to use it was the 1978 Atari 400, while the last use was on the XE series, which ended its commercial cycle in the late 1980s. For its VCS console (later known as 2600) it used the 6507, a cheap version of the 6502 with a package with a small number of external addressing lines and therefore capable of handling only 4 KB of memory, to keep production costs low.

The English Acorn Computer used the 6502 for its 1980 Acorn Atom and for the much more famous BBC Micro of 1981, which remained on the market for 13 years and used by the BBC both for a computer television course and in British schools.

Commodore was one of the main users of 6502, and we would miss it too, since the factory that produced it was his! The 6502 was used in Commodore PET first and VIC-20 then. Many of its derivatives were also used in the Commodore house: the 6509, used in the little widespread CBM-II computer line; the 6510, used only in the famous Commodore 64; the 7501, used on C16 and Plus/4; the 8502, for the Commodore 128. The 6502 was also used as a processor in Commodore disc drives to run the DOS which, unlike other home computers, was contained in a ROM of the player itself and run independently of the main computer software.

Nintendo built a custom CPU for its NES (Nintendo Entertainment System) console derived from 6502, the Ricoh 2A03/07 (NTSC/PAL), where the BCD arithmetic management logic makes room for a programmable sound generator.

Numerous arcade games also used the 6502, both as a main processor and as a coprocessor. Among the best known are Super Breakout, Burger Time, Missile Command and the famous Asteroids.
Finally, how can we not mention the 6502 used as Bender's brain processor, the main robot in the cartoon series "Futurama"!

WDC 65C02 and 65C816

After the acquisition by Commodore, some MOS engineers decided to leave the company. Among them was Bill Mensch, who founded the Western Design Center (WDC) in 1978. Having acquired the rights to the 6502, he revised the project by correcting the bugs in the original CPU and, in addition, decided to use the new CMOS production technology and also to make the chip completely static, that is to say that the clock could be stopped completely without the processor losing the data currently present in its registers.

The WDC 6502 was used on various computers, such as the Apple IIc and Apple Enhanced IIe, the BBC Master

Fig.7 - Bill Mensch, co-designer of 6502 and founder of Western Design Center (photo by Jason Scott)

(heir to the BBC Micro) and the Laser 128 (8-bit Atari clone). It was used in the Atari 5200, released in 1982 as heir to the 2600, and in the 7800 of 1986, as well as in the Atari Lynx of 1987, one of the first portable consoles, and in the XEGS (derived from the Atari XE) of 1987. The late 1980s PC-Engine/Turbografx-16 mounted the Hudson HuC6280, an integrated derived from the WDC 65C02 with several additions, such as a memory manager, a timer, a parallel port, and more. WDC continues to produce and market 65C02, which has remained on the market for over 40 years.

Another product of WDC derived from 6502 is 65C816. This processor is essentially a 16-bit evolution of the processor that allows, however, to also execute 8-bit code thanks to a particular software-activated emulation mode (hence the number 816 in the acronym). It was used as a CPU in the Apple IIGS, allowing this computer to run both the 16-bit programs written for it and the old 8-bit software written for earlier Apple II models based on the 6502.

CONCLUSIONS

The 6502 was a processor that made the history of computing and contributed, thanks to its economy, to the spread of cheap computers among the masses.

Let's not forget that the first computer built by Steve Wozniak became a reality thanks to the fact that the 6502 was put on the market at an affordable price even by a student who was a bit on the bill what was the legendary "Woz".

Or that all 8-bit Commodore computers, including the famous C64, were based on a 65xx family CPU. Not to mention that the emergence of home consoles was achieved thanks to very important and widespread gaming systems such as the Atari 2600 before and the "Nintendo" (or NES) after, both based on CPUs derived directly from the 6502.

Structuring old BASIC dialects with FOR-NEXT loops - part 2

by Alberto Apostolo

The papers presented about this topic (RMW 27-ITA/05-ENG for ZX Spectrum, RMW 28-ITA/06-ENG for other computers) opened a debate in our readers. Many interesting posts inspired me to write a sequel.
The idea of creating structures with FOR-NEXT loops was essentially appreciated, despite some criticism for the difficulties of writing programs and for the slowing down of the execution speed (from 10% to about 20%).

In BASIC versions that do not support STEP clause or do not allow "STEP 0" option, you can make a loop virtually infinite by placing

```
FOR F = 0 TO N
```

where N = 1E+38 (or at least a very large number).
In addition, it has been highlighted that some One-Liners handle conditional expression in the TO clause of FOR-NEXT loops ("THEN" branch in Fig.1).

```
REM IF
          LET A = condition
REM THEN
          FOR F = 1 TO A

              then_statements

          NEXT F
REM ELSE
          FOR F = A TO 0

              else_statements

          NEXT F
REM END-IF
```

Fig. 1

The BREAK statement

Some readers have expressed the need to have an equivalent of the "break" command (present in the C language) in order to exit a loop immediately.
Slavo Labsky (reader of RMW ENG) sent the Facebook group "BASIC ON THE ZX SPECTRUM" an elegant suggestion (Fig. 2) based on an appropriate change in the count of FOR command.
Unfortunately, the idea for the ZX Spectrum may not work on other versions of BASIC. For example, in GW-BASIC the error "FOR WITHOUT NEXT" is encountered (Fig.3).

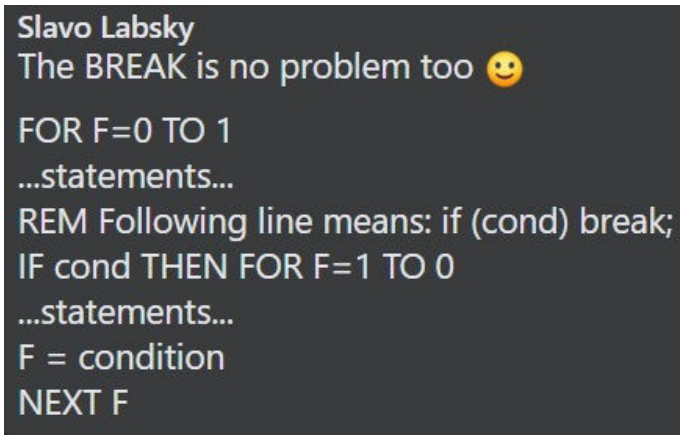


Fig. 2

```
1000 REM THIS "WHILE" STRUCTURE
1005 REM DOESN'T WORK IN GW-BASIC
1010 REM "FOR WITHOUT NEXT AT LINE 2010"
1015 REM OCCURS
2000 X = 10
2010 FOR F = 0 TO 1 STEP 0
2020 IF NOT(X<=30) THEN FOR F = 1 TO 0
2030 INPUT X
2040 PRINT X
2050 NEXT F
2060 STOP
```

Fig. 3

The CASE-WHEN structure

In Fig.4 I propose a possible solution suitable for many variants of BASIC, but everyone is free to invent alternative solutions.

Conditional expressions in arithmetic calculations

Not all variants of BASIC have the versatility of BASIC Sinclair and it is not always permissible to write instructions of the type

```
LET X = X + ( A$="8" AND X<= 31)
```

If you are dealing with numerical variables, you can define some work functions with a DEF FN command (Fig.5) and get to write

```
X=X+FN WAND(FN WEQ(ASC(A$),ASC("8")),FN WLE(X,31))
```

It is advisable to apply the functions listed in Fig.5 to integer variables because with floating-point variables, unpleasant effects due to rounding may occur (in some particular cases).

Conclusions

The trick has been discovered by now.
In BASIC you can unconventionally use FOR, NEXT, GOSUB statements to eliminate the use of "GO TO" and transform a BASIC version (poor of instructions) into a structured language.

The price to pay is greater attention in writing programs and endure a certain slowdown in execution.

```
              CASE
                   WHEN cond_1 THEN [statements_1]
                   WHEN cond_2 THEN [statements_2]

                   . . .

                   ELSE [else_statements]
              END

 100 REM main program

    . . .

 499 REM CASE
 500           LET A = 0
 510           IF cond_1 AND A = 0 THEN LET A = 1: GOSUB 1100
 520           IF cond_2 AND A = 0 THEN LET A = 2: GOSUB 1200

               . . .

 598           IF A = 0 THEN GOSUB 1990
 599 REM END

     . . .

 999 STOP

1100 REM branch 1
                   statements_1
1199 RETURN

1200 REM branch 2
                   statements_2
1299 RETURN

     . . .

1990 REM else
                   else_statements
1999 RETURN
```

Fig.4

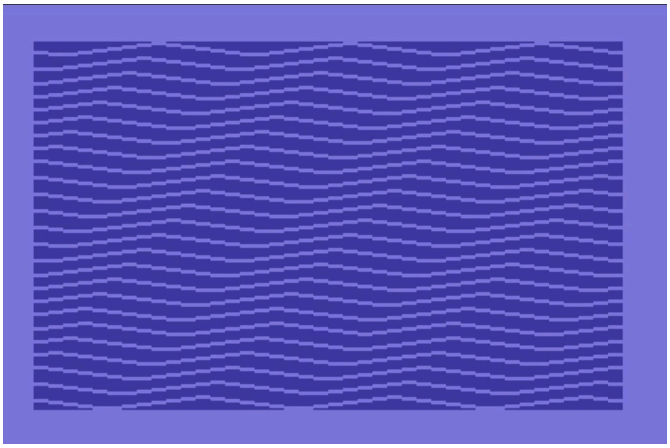
```
DEF FN WNE(X,Y)  = SGN(ABS(X-Y))        : REM X <> Y
DEF FN WGE(X,Y)  = SGN(1+SGN(X-Y))      : REM X >= Y
DEF FN WLE(X,Y)  = SGN(1-SGN(X-Y))      : REM X <= Y
DEF FN WEQ(X,Y)  = 1 - SGN(ABS(X-Y))    : REM X = Y
DEF FN WLT(X,Y)  = 1 - SGN(1+SGN(X-Y)) : REM X < Y
DEF FN WGT(X,Y)  = 1 - SGN(1-SGN(X-Y)) : REM X > Y
DEF FN WNOT(X)   = 1 - X
DEF FN WAND(X,Y) = X * Y      : REM WAND3(X,Y,Z) = X*Y*Z
DEF FN WOR(X,Y)  = SGN(X+Y)   : REM WOR3(X,Y,Z)  = SGN(X+Y+Z)
DEF FN WXOR(X,Y) = X*X - Y*Y : REM WXOR(X,Y)    = ABS(X-Y)
DEF FN WMIN(X,Y) = ( X + Y - ABS(X-Y) )/2
DEF FN WMAX(X,Y) = ( X + Y + ABS(X-Y) )/2
```

Fig.5

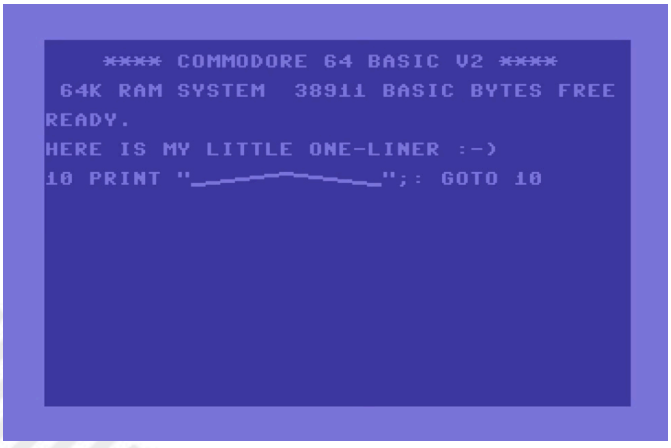
BASIC in a nutshell: waves on LM80C and MSX-1

by Antonino Porcino

Recently I happened to come across a nice 1-liner program (i.e. made from a SINGLE BASIC line) for Commodore 64 that creates the illusion of a swaying surface [1].



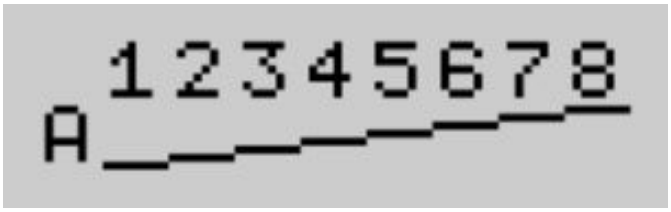
The list below is easy to explain: using the PETSCII graphic characters of the C64, PRINT prints a piece of the wave that occupies 13 characters. Three waves are 13 x 3 = 39 characters long and since the C64 screen is 40 columns long, the missing character creates a phase shift on the next line, thus drawing inclined curves. Then scrolling the screen up creates the illusion of movement.



A simple but ingenious program that aroused my curiosity, so I told myself, how to adapt it to the LM80C?

For those who do not know it, the LM80C is the computer entirely created by Leonardo Miliani and presented on the pages of this magazine in previous issues. It is based on the Z80 processor and TMS9918 video chip; it also has a powerful BASIC interpreter customized to take advantage of the machine's graphics and sound potential. From time to time I enjoy writing small BASIC listings for this computer using one of the emulators I have created: the software emulator for the browser [2], or the FPGA implementation for MiST [3].

How can this 1-liner programme be adapted to the LM80C? Thelatter does not have those particular PETSCII characters used in the list. These are 8 characters whose shape is simply a horizontal line; for each of the 8 characters the line rises by one line:



In the LM80C, but also on any computer that uses the TMS9918 chip, you can redefine the characters by writing to the video RAM, modifying what is called the "Pattern Generator Table". You can easily do this from BASIC with the VPOKE command, which is identical to normal POKE only writing to video memory instead of normal RAM. This is because the TMS9918 chip has its own dedicated RAM, separated from the system memory.
For simplicity I decide to redefine the graphics of the characters “abcdefgh” (in lowercase).

In TMS9918 the "Pattern Generator Table” starts at address 0, so the lowercase "a” graphics starts from address 97 x 8, where 97 is the ASCII code of the "a” and 8 are the number of lines per character (the character array is the classic 8x8).

Then I write:

```
20 S=97*8
```

I then clean the 8 characters starting from the "a”, writing the value “0” with VPOKE:

```
30 FOR T=0 TO 64
40 VPOKE S+T,0
50 NEXT
```

Now I have to draw the horizontal lines of all eight characters. I start by moving to the eighth line of the letter "a" with

```
60 S=S+7
```

Horizontal line graphics is nothing more than the value 255, that is, all the bits on. I also notice how to move to the next character I should do S=S+8 but I also have to go up one line, so S=S+7.

Ultimately:

```
70 FOR T=0 TO 7
80 VPOKE S+T,255
90 S=S+7
100 NEXT
```

Now that I've redefined the characters, I can finally write a print loop similar to the original C64 program:

```
110 PRINT "aabcdefghhgefbcda";
120 GOTO 110
```

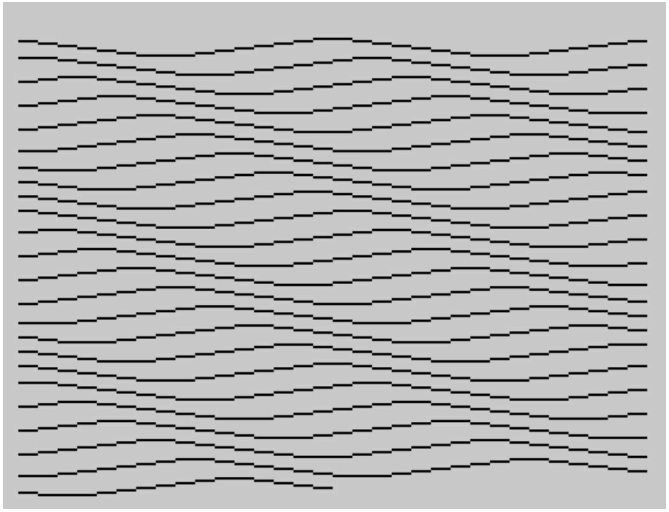
Notice how I repeated “a” and “h” twice, this to arrive at a string of 17 characters. Two strings reach 34 characters creating a two-position offset from the 32 columns of the LM80C screen.

Here is the complete list:

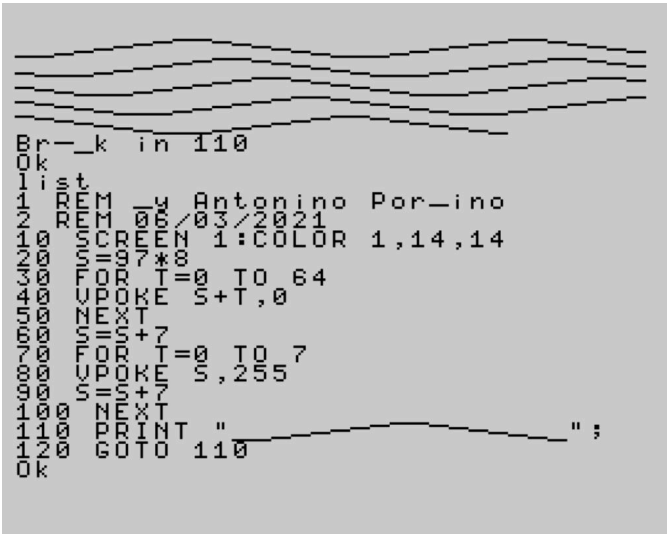
```
     LM80C 64K Color Computer
 by Leonardo Miliani * FW R1.05

   <C>old or <W>arm start? W
Ok
LIST
1 REM by Antonino Porcino
2 REM 06/03/2021
10 SCREEN 1:COLOR 1,14,14
20 S=97*8
30 FOR T=0 TO 64
40 VPOKE S+T,0
50 NEXT
60 S=S+7
70 FOR T=0 TO 7
80 VPOKE S,255
90 S=S+7
100 NEXT
110 PRINT "aabcdefghhgfedcba";
120 GOTO 110
Ok
```

Which produces the following result visible in the next image. To appreciate the effect of the animation (not reproducible on the pages of a magazine) you can simply click on the link [4] of the online emulator:
https://nippur72.github.io/lm80c-emu?load=prg/wave.prg



If you stop it with the ESC key (which acts as A BREAK), you can see that in the list the characters “abcdefgh” appear redefined:



This lasts until the computer is warmly reset, either with the PHYSICAL RESET button or with THE Basic command of the same name (the combination CTRL+ALT+PAUSE also applies on the emulator). I conclude this short article by thanking Orazio Cacciola who pointed out that the program also works for MSX-1 without too many changes; in fact, with the LM80C they have the same video chip and the same VPOKE command in common. In the link [5] you will find the video running on MSX-1.

[1] Video of the original one-liner program:
https://www.youtube.com/watch?v=0yKwJJw6Abs&ab_channel=JuhuTuubi
[2] The LM80C emulator:
https://nippur72.github.io/lm80c-emu
[3] FPGA implementation of the LM80C:
https://github.com/nippur72/LM80C_MiST
[4] Program WAVE.bas running on the emulator:
https://nippur72.github.io/lm80c-emu?load=prg/wave.prg
[5] Port version for MSX-1 by Orazio Cacciola:
https://www.youtube.com/watch?v=_AmkKIA24Dc&ab_channel=oraziocacciola

Grapheur 1.0 - Graphs with the Amstrad CPC

by Francesco Fiorentini

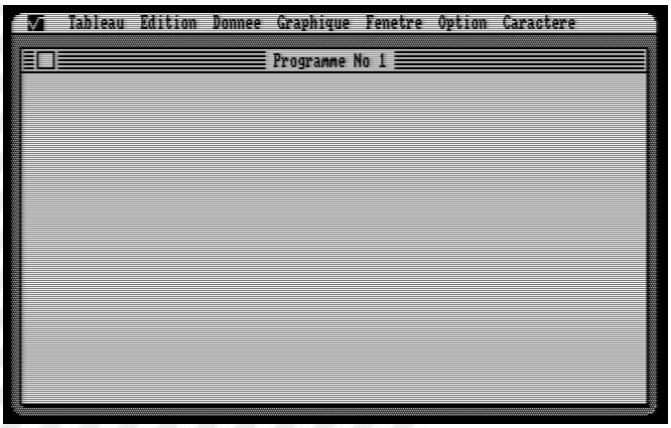
Those who follow my articles on RetroMagazine World know that every now and then I like to review, in addition to games, 8-bit and 16-bit application software.
This passion stems from the desire to understand how these machines were, or could be, used for professional or semi-professional purposes before the almost unconditional surrender to the MS Office suite.

The software I present to you this time is called Grapheur and was distributed in France by Micrologic and E.S.A.T. Software. Amstrad CPC did not have much luck in Italy, where the market was dominated by Commodore and Sinclair, but in France and Spain this computer had a completely different impact on the market back in the '90s.

Launch the program using the command:

LOAD "GRAPHEUR.BAS"

Once the upload is complete and the initial splash screen passes, we are faced with a pleasant and very professional graphical interface. Well done!



Being a French software, developed by **Patrick LEFEBVRE**, the menus are obviously in this language and without the possibility of localization, but fortunately, also with my basic French, I can easily master the program.

Before we move on to drawing a chart, we need to create a table that contains the data that will be represented graphically. As with databases, before creating the data structure it is good to keep in mind what we want to represent, in order to avoid subsequent changes that could undermine the stability of the structure.
So I decide to represent this table that I found on the web.

Sport practiced	Males	Females
Soccer	19	1
Gymnastics/ Fitness	3	9
Swim	7	8
Dance	0	9
Volleyball	2	8
Martial arts	3	1
Other sports	4	0
Any sport	12	14
Totals	50	50

From the Tableau menu, choose **Ouvrir** and build an 8-row table with 2 columns:

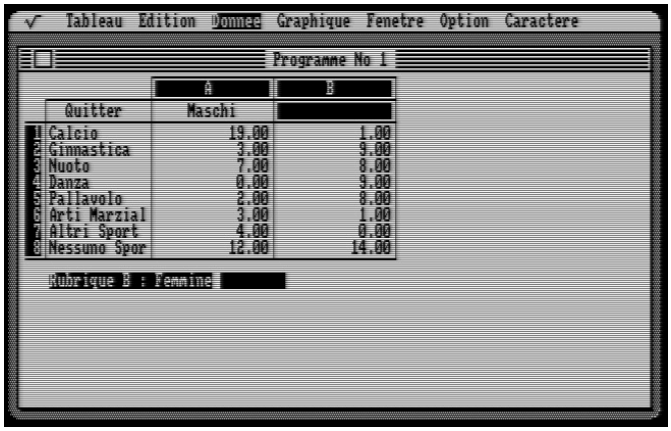
Tableau Edition Donnee Graphique Fenetre Option Caractere
Programme No 1
Ouvrir
Les donnees sont: 1) Litterales
2) Mensuelles
3) Annuelles
Votre choix (1 a 3) : 1
Nb de rangee (1 a 48) :8
Nb de rubrique (1 a 4) :2

Then the table will be displayed automatically and we have to go to **Donnee** (Data) and select **Saisir** (Insert) to enter the values until they are completed.

Remember that you can use these commands to move easily between the program menus and within the tables:
- **Cursor keys** to move between entries

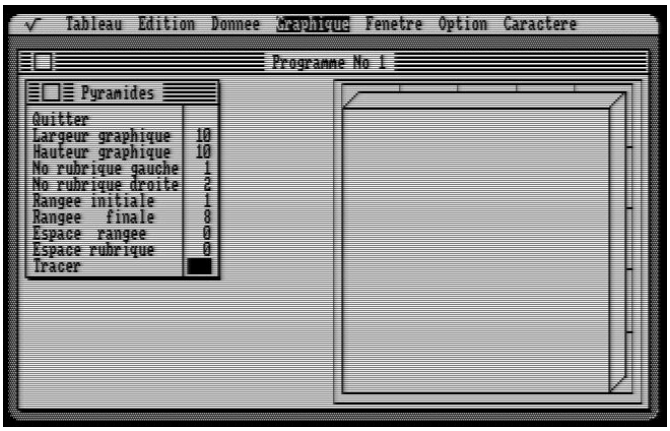
- **Space** to select a menu item
- **Return** to confirm the value entered
- **ALT** to enter the alternative menus (e.g. Quitter in the table)
- **Quitter** to exit drop down menus

Once all the data has been entered, we will enter the column headings; then our table should look like this:



Now that we have the data in our chart, we can try to create a chart.

From the **Graphiqué** menu I choose Barres, but after populating the options and choosing Tracer to draw the graph, an incomprehensible drawing appears. I get the same result by choosing Histogrames... I have to admit I'm a little frustrated by this lack of results. :- (

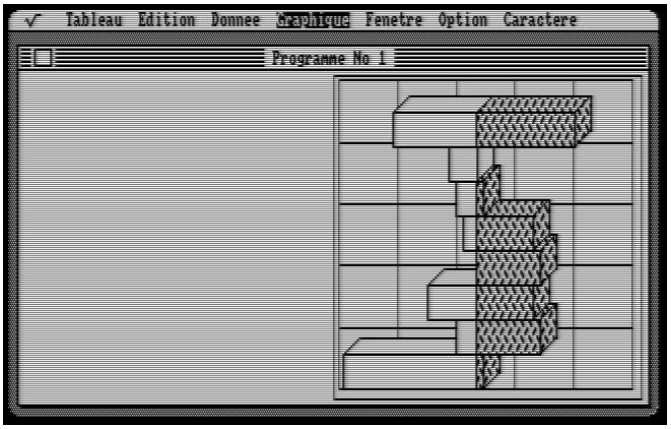


The unavailability of the manual in this case is a problem. However, it will certainly not be a first failure to stop me; I continue to study the menus without, however, finding the key to displaying bar graphs and histograms satisfactorily. So I decide to try a new chart type and choose Pyramides.

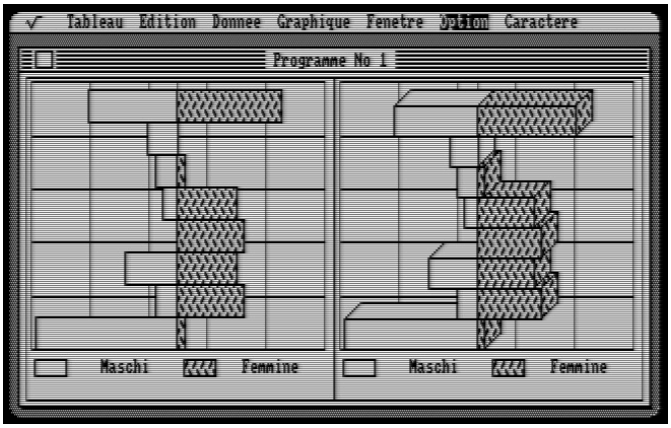
This time in the chart options I notice a couple of entries that remind me of something and then try to populate the entries accordingly:

- **Nr. rubrique gauche**: 1
- **No droite heading**: 2
- **Initial range**: 1
- **Final range**: 8

I click on **Tracer** and... drum roll... my chart is finally drawn:



Satisfied by the result obtained, I continue to play a little more with the menus, being able to view the legend and duplicate the graph creating one in 2D and one in 3D:



Conclusions

The program is full of options and possibilities. Probably when it was released, accompanied by his manual, it was a very respectable program; it was not by chance sold for a price of about 300 francs, not exactly a gift. Currently, however, it is relegated to the sole purpose of preserving the software; useful to demonstrate how an Amstrad CPC could be used in the same way as a personal computer in a small office.

Unfortunately, the lack of the manual makes it incomplete for preservation purposes. If anyone has the manual, contact us, or contact CPC-Power.

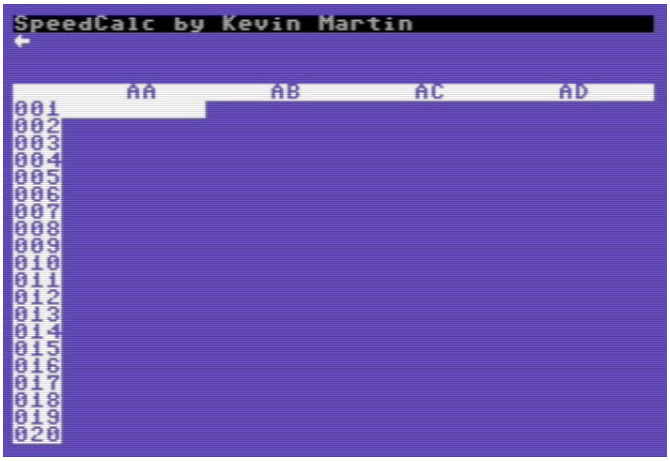
Grapheur can be downloaded from **CPC-Power**:

https://www.cpc-power.com/index.php?page=detail&num=13866

SpeedCalc 2019 - is it Excel on C64? Well, almost...

by Francesco Fiorentini

After drawing graphics with Amstrad CPC, I want to remain in the office area, but change computers. I found this interesting spreadsheet for Commodore 64 that I think is worth reviewing.



SpeedCalc 2019 is an advanced version of the 1986 SpeedCalc by Kevin Martin. Since it is a spreadsheet, Excel immediately comes to mind, although, in this case, the most obvious similarity is with Lotus 123. Older readers will certainly remember the Lotus Software spreadsheet and will have no difficulty comparing it to SpeedCalc.

The workspace

The SpeedCalc workspace is divided into 3 parts. The first line starting from the top is reserved for messages. Rows 2 through 4 are the workspace, where the user enters data and formulas. The remaining 21 rows are the actual spreadsheet.

The spreadsheet consists of 200 rows for 50 columns, for a total of 10,000 viewable cells! Unfortunately, due to obvious memory limitations, only a part (about a third) of the cells can contain data.

The controls

The list of commands available in SpeedCalc is relatively long. Some of these commands can be invoked by pressing a single key, while for others it is necessary to press 2 or even 3 simultaneously. Let's see them:

CTRL-A: Displays available memory
CTRL-B: empty (delete) the current cell
CTRL-C: Copy the cell or group of cells
CTRL-D: Sets the number of decimals
CTRL-E: Modify the current cell
CTRL-F: Change cell format
CTRL-G: Go to the selected cell
CTRL-M: Moves a cell or group of cells
CTRL-P: Print Sheet
CTRL-R: Turn automatic recalculation on/off
CTRL-W: Change column width
CTRL-X: Quits SpeedCalc (asks for confirmation)
CTRL-4: Disk directory
CTRL- ^: Send disk command
CLR / HOME: home cursor
SHIFT-CTRL-C: Relative block copy
SHIFT-CTRL-D: Change decimal mode for all cells
SHIFT-CTRL-M: Move relative block
SHIFT-CTRL-P: screen, disk, or printer printing
SHIFT-CTRL-R: Displays the current recalculation mode
SHIFT-CTRL-W: Change the width of all columns
SHIFT-CLR / HOME: erase entire sheet
F1: change border color
F3: Change the background color
F5: change font color
F7: Upload SpeedCalc file
F8: Save SpeedCalc file
<-: recalculate sheet

Note*: For THOSE who use VICE, the CTRL key is mapped to the TAB key.*

Fig. 1 - Justified text and modified colors

This list, however, is not exhaustive.

I noticed that it is possible to justify the text contained in the cells by using THE CTRL-~ command and then specifying L for left, C for Center and R for Right, see example in Fig. 1.

Functions

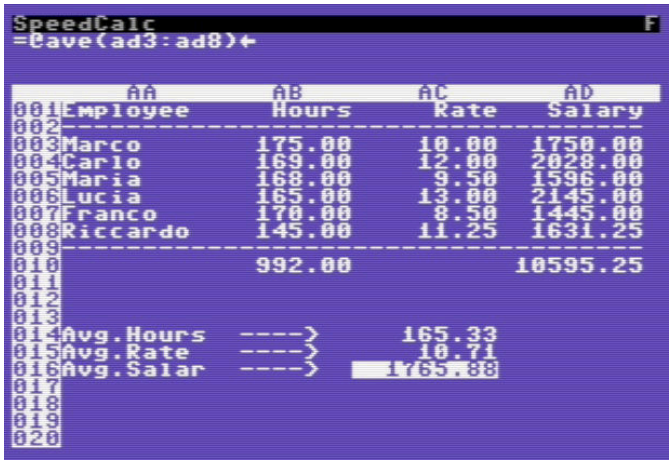
@abs () absolute value
@atn () arctangent
@ave () average of a cell block [form: @ave (xxn: xxn)]
@cos () cosine of the argument in radians
@exp () logarithm complement, returns and^X
@int () integer (rounds to the lowest integer)
@log () natural logarithm based on and (log <=0 is illegal)
@sgn () sign (-1 for negative numbers, 0 for 0, 1 for positive)
@sin () argument sine in radians
@sqr () square root (sqr of a negative number is illegal)
@sum () sum of a block of cells [form: @sum (xxn: xxn)]
@tan () argument tangent in radians
@pi value of pi (3.14159265)

Road test

Now that we have all the information we are able to start using this software productively. Let's try to create a sheet with some simple calculations: multiplications, sums and an average.

I try to put myself in the position of a hypothetical employer of the 1980s who has to pay his employees according to the hours worked and needs to know quickly how much the total to pay.

To make the example more readable and close to the present day, I have expressed monetary values in not specific currency (euro or dollar, you choose...).



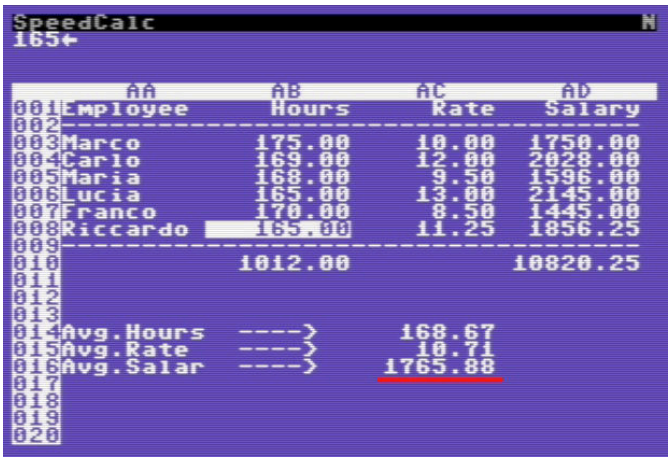
Those who are used to working with modern spreadsheets will have to get a little used to the new commands and especially to the fact that they cannot change the values already entered except by using the appropriate command CTRL+E.

For the rest, I have to admit that after an initial struggle, the management of the tool is, after all, straightforward. You can copy cells, move them, and even make copies of them, just like you would in Excel.

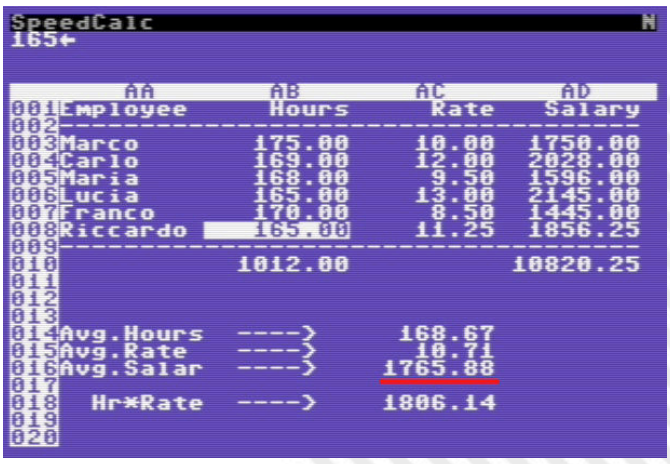
The thing that struck me in the negative, and that I had not noticed during the writing of the first part of the article, is the lack of some basic functions that I would have considered useful. I refer to functions to find the minimum or maximum value within a cell block (for example). Surely these functions are much more useful in the office environment than the trigonometric ones. Pretty weird.

An annoying bug

Unfortunately, I have to tell you that in my brief rehearsal I ran into a small but annoying bug. By varying the amount of hours worked by Richard, the sheet is automatically recalculated (I turned on automatic recalculation using CTRL-R), but I noticed that a formula is not updated. I do not think that this depends on the formula because it is correct, but rather on a bug in the program.



I tried to edit other cells, too, but the problem remains. I also added another formula (which is recalculated correctly), but the bug of not recalculating average wages remains. By placing us in the cell and forcing a recalculation from there, the value is updated.



Too bad, a great tool penalized by an annoying bug!

SpeedCalc 2019 can be downloaded from:
http://cbm.ficicilar.name.tr/program/27/speedcalc-2019

Simulating PLOT X,Y in C64 bitmap mode

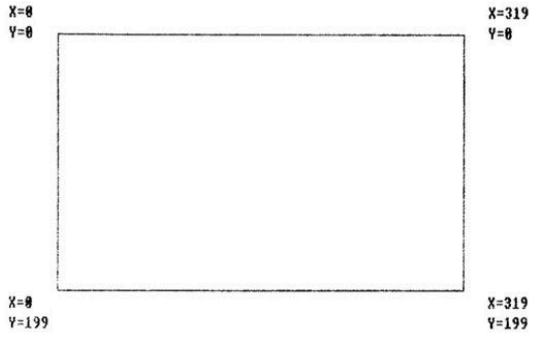
by Attilio Capuozzo Founder of RetroProgramming Italia – RP Italia

Before moving on to the theme of this episode, let's say that here we will omit to explain how to activate a Bitmap (the Graphics Page) and how to tell the VIC-II graphics chip what the start address of the Bitmap allocation is. For these topics we refer you to the excellent articles by **Tonino Scaffidi** "**Grafica HiRes e Asm con il C64**" Part #1 and #2 published on the **RETROPROGRAMMING ITALIA (RPI) group**.

The Standard Bitmap Mode, also referred to as HiRes or High Resolution Graphics, is the alternative mode to the C64 Character Mode and involves working on a 320*200 pixel screen.
Since each pixel is encoded in 1-bit memory, it is easy to calculate that the Graphics Page occupies 64,000 bits of memory corresponding to 8,000 bytes (64,000/8).

So in memory we will need to allocate a space of 8K that will necessarily correspond to the 1st half or 2nd half of the 16K bench to which the VIC-II can access; the 16K memory banks have been extensively explained in **Attilio Capuozzo**'s series of articles "**How to Create Custom Characters on the C64**" published as well on **RETROPROGRMMING ITALIA (RPI)**.

Leaving aside the operational details of how to allocate storage space for the Bitmap (see the aforementioned "HiRes and ASM Graphics with C64"), let's see how we can simulate THE PLOT on the Graphics Page to "turn on" a single pixel of X and Y coordinates, taking into account that the Bitmap has a horizontal resolution of 320 pixels and a vertical resolution of 200 pixels (see figure).



The Bitmap can be imagined as a Page consisting of 1,000 Programmable Characters distributed over 25 Rows of 40 Columns each, a bit like in a Text Display of the C64 Character Mode.
Looking at Fig. 2, attached to the article, we can easily see that each character occupies a grid of 8*8 pixels (i.e. 8 bytes in height) and since in a Row there are 40 Columns it is easy to calculate that each Row contains a total of 320 bytes (40*8) arranged according to the particular sequence shown in Fig. 2.

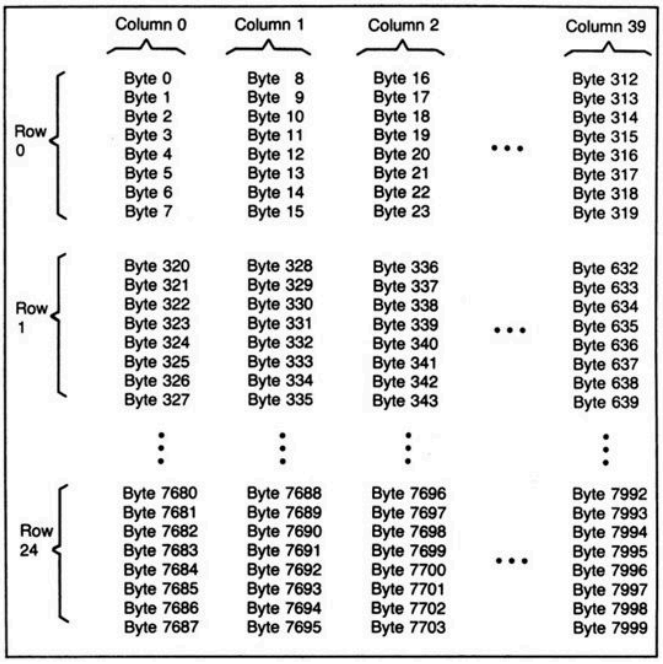

Column 0
Column 1
Column 2
Column 39
Row 0
Byte 0 Byte 1 Byte 2 Byte 3 Byte 4 Byte 5 Byte 6 Byte 7
Byte 8 Byte 9 Byte 10 Byte 11 Byte 12 Byte 13 Byte 14 Byte 15
Byte 16 Byte 17 Byte 18 Byte 19 Byte 20 Byte 21 Byte 22 Byte 23
Byte 312 Byte 313 Byte 314 Byte 315 Byte 316 Byte 317 Byte 318 Byte 319
Row 1
Byte 320 Byte 321 Byte 322 Byte 323 Byte 324 Byte 325 Byte 326 Byte 327
Byte 328 Byte 329 Byte 330 Byte 331 Byte 332 Byte 333 Byte 334 Byte 335
Byte 336 Byte 337 Byte 338 Byte 339 Byte 340 Byte 341 Byte 342 Byte 343
Byte 632 Byte 633 Byte 634 Byte 635 Byte 636 Byte 637 Byte 638 Byte 639
Row 24
Byte 7680 Byte 7681 Byte 7682 Byte 7683 Byte 7684 Byte 7685 Byte 7686 Byte 7687
Byte 7688 Byte 7689 Byte 7690 Byte 7691 Byte 7692 Byte 7693 Byte 7694 Byte 7695
Byte 7696 Byte 7697 Byte 7698 Byte 7699 Byte 7700 Byte 7701 Byte 7702 Byte 7703
Byte 7992 Byte 7993 Byte 7994 Byte 7995 Byte 7996 Byte 7997 Byte 7998 Byte 7999

Fig. 2 - Chars per row

Our purpose is therefore to find a mathematical formula that allows us to calculate the absolute address of the byte and - within the scope of the byte found – the bit that corresponds exactly to the pixel to be turned on of X and Y coordinates.

It must be said that the calculation may appear at first glance rather complex or otherwise cumbersome, but if we can carefully follow the logic that leads us to the mathematical formula everything will appear clearer to us! We indicate with X and Y respectively the Horizontal and Vertical coordinate of the pixel to be turned on.

We will first calculate the Vertical and Horizontal Offsets of the byte to which we will then add the start address of the Bitmap to find the Absolute Address of the byte.

Finally, we will trace the position of the bit within the byte. Let's start by calculating the vertical information of the pixel: we said that each Character on the Bitmap is represented by a square area of 8*8 pixels so to find the Line number (from 0 to 24) just divide the Vertical Y coordinate by 8 (we will only take the whole part of the division):

RI = INT(Y/8)

To know the number of the start byte of the RI Line found

before, simply multiply RI by 320 (Line 0 starts with byte 0, Line 1 with byte 320... Line 24 with byte 7680- see Fig. 2):

start bytes Line = RI * 320

Now we have to calculate the Line within the Row (each Row is formed in height by 8 Lines corresponding to 8 bytes – see Fig.2): to obtain the Line number (from 0 to 7) simply calculate the rest of the division of the Vertical position Y by 8.
To obtain the rest of a division for a power of 2 (in this case 8 which is equivalent to 2^3) simply put in AND the number in question (Vertical coordinate Y) with the power of 2 -1 that is with 7:
LI = (V AND 7)

Then by combining the above calculations together we will obtain the total Vertical Offset of the byte in which our pixel is positioned:
VO = INT(V/8) * 320 + (V AND 7)

Now let's move on to calculating the Horizontal Offset of the byte:
to know the number of Column (from 0 to 39), simply divide the horizontal X coordinate of the pixel by 8; again we will consider only the entire part, omitting any decimal fraction:
CO = INT(X/8)

Note: as for the calculation of the Row number, even in the case of the Column number we must divide by 8 because, as mentioned, the 1,000 Programmable Characters that conventionally form the Graphic RAM are contained in 8*8 pixel grids that in the Bitmap Mode are called Cell and that are roughly similar to the Character Block of the Text Display in the Character Mode.
If we look closely at Fig. 2 we notice that, due to the particular arrangement followed by the bytes within the Bitmap, the start byte of each column is exactly 8 bytes away from the one preceding it (Row 0: Column 0-->byte 0, Column 1-->byte 8, Column 2-->byte 16 and so on). Therefore, if we multiply the Column number * 8, we easily calculate the total Horizontal Offset (that is, the start byte of the Column number calculated previously):
HO = INT(X/8) * 8

Now by adding together the Start Address (i.e. the initial memory location) of the Bitmap BA with the Vertical VO and Horizontal HO Offsets we will finally find the Absolute Address of the byte BY within which the bit corresponding to the pixel to be turned on on the screen is located:
BY = BA + VO + HO
We have almost reached the end...all we have to do is calculate the bit that corresponds to our pixel; we resort again to the use of the logical operator AND to find the rest of a division for a power of 2 or 8 (2^3):

PX = (X AND 7)

We took the Horizontal X coordinate of our pixel and putting it in AND with the power of 2 -1 (ie 8 – 1) we calculated the rest of the division between X and 8 (8 is the number of pixels in each Column equal to the number of bits in each byte).
The remainder PX corresponds to the number of pixels to the left of our pixel within a group of 8 consecutive pixels constituting a specific Column.

The problem is that the horizontal positions of the pixel on the screen go from left to right while the bits in a byte are numbered in reverse, i.e. proceeding from right to left (bit 0, the 1st bit, is the rightmost bit and is defined as LSB=Least Significant Bit; bit 7, the 8th bit, is the last leftmost bit and is also called MSB=Most Significant Bit).

Therefore, to calculate the BI bit, just do the inverse of the above calculation (the one related to PX):
BI = 7 – (X AND 7)

Perfect! With bytes and bits in hand we can finally use the instruction to “turn on” the pixel of X and Y coordinates, i.e. set the corresponding bit in memory to 1:
POKE BY,PEEK OR (2^BI)

With the aforementioned instruction the pixel will take on the Color whose numerical Code (from 0 to 15) is contained in the High Nibble (bits from 4 to 7) of the corresponding byte of the Screen Memory which, in Bitmap Mode, becomes a Color Map of the Graphics Page (see the aforementioned articles “HiRes and ASM Graphics with the C64”).

In the same way we can “turn off” the pixel i.e. set the bit to 0 with the instruction:
POKE BY,PEEK(BY) AND (255-2^BI)

In this case the pixel will take the Color whose numerical Code (always from 0 to 15) is contained in the Low Nibble (bits from 0 to 3) of the Screen Memory.

That's all folks!

RetroProgramming Italia - RP Italia:
https://www.facebook.com/groups/retroprogramming/

May the FORTH be with us - part 3

by Francesco Fiorentini

After taking a look at the vocabulary and stack, it's time to see how we can create constants and variables to use in our programs.

Constants

To define a constant we must use the constant keyword:

```
10 CONSTANT TEN
```

this command creates the constant TEN with the value 10.
To see the value of the constant simply type the following command:

```
TEN .
```

and the Jupiter ACE will answer you politely:

```
TEN . 10 OK
```

Constants can also be redefined using the REDEFINE command:

```
11 CONSTANT TEN REDEFINE TEN
```

Numerical variables

In a very similar way the variables can be declared:

```
15 VARIABLE VAR1
```

in this way we will initialize the variable VAR1 with the value 15. We would expect with the command:

```
VAR1 .
```

to see the value 15 printed on the screen, instead our computer returns a seemingly meaningless value.
In fact, the value returned from the computer is nothing more than the memory location where our variable is stored.

To print the value of THE VAR1 variable on screen we must use the **@ (fetch)** command:

```
VAR1 @ .
```

This will print the value 15 correctly on the screen. Weird? Maybe, but by now we should be used to the quirks of the Forth... :-)

Now let's try using the constant and the variable in a simple program:

```
: Squares
ten ten * .
var1 @ dup * .
;
```

Probably to those who are used to the syntax of Basic or other languages, you will find this list a bit difficult, but if you have carefully followed the previous two lessons it should seem relatively familiar.
This funny 'word' prints on screen the square of the constant TEN and the variable VAR1 just created.

Strings

Typically programming languages allow you to manage strings with native functions. The ACE Jupiter Forth, on the other hand, does not have string management within its vocabulary. Fortunately, this is not a problem, since we can define new commands ourselves.

A banal way to create and initialize a string can be simulated with the following word:

```
: msg1. " Message" ;
```

In this way we have created the word msg1 that contains nothing more than the string "Message" inside it.
Obviously, to print the message on screen, simply recall the word using:

```
msg1
```

This system, however, is decidedly simplistic and does not allow us to manipulate strings as we are used to doing with other types of languages.
Fortunately, as mentioned above, we can define A TYPE OF STRING data that allows it to be manipulated.
Then enter the following code:

```
: count
dup 1+ swap c@
;
define string
ascii " word count dup c,
over + swap
do
i c@ c,
loop
does>
count ;
```

Now try to declare a string using:

```
string msg Hello world by RMW"
```

and then try printing it using the type command:
msg type
To prove that it can be manipulated, we implement FOR EXAMPLE THE MID command:

```
: mid
swap 1 max 3 pick min 1-
swat rot min over max
over - rot rot + swap ;
```

and let's use it:

```
msg 1 5 mid type
```

That'll be all for today, see you in part 4 of the next issue.

BASIC 10-LINER CONTEST 2021

by Alberto Apostolo

On April 10, 2021, as part of the NOMAM (Not Only Marvelous Atari Machinery) event held annually in Germany, the award ceremony of the "BASIC 10Liner Contest 2021" (Fig.1) organized by Gunnar Kanold (Fig.2) and his friends from the Homeputerium group took place.

Fig. 1

Homeputerium is a laboratory located within the Eider-Treene-Schule in Friedrichstadt, Germany. In the laboratory, students can practice with home-computers from the 1980s donated to the School.
As the name says, "BASIC 10Liner Contest" means trying to create (documented) BASIC language program no longer than 10 lines (in [Kan21b] you can find the link to the rules, written in German and English).
Participations (from all over the world) were received between February 1, 2021 (when the post appeared on Twitter) and 6:00 pm on March 27, 2021 (deadline for admission).
The 146 programs registered for the 2021 edition have been divided into five categories:
1) PUR-80 (games in 10 lines, max 80 chr. per line),
2) PUR-120 (games in 10 lines, max 120 chr. per line),
3) EXTREME-256 (games in 10 lines, max 256 chr. per line),
4) SCHAU (=show i.e. utility/demo/applications in 10 lines, max 256 chr. per line),
5) WILD (programs that do not respect the previous categories).
The competition was fierce (everyone could send more programs for the race) and a very strict jury had to do a difficult job.

Fig. 2

The HIGHLIST program

I usually follow this wise advice: participate in contests only if you already have something that you think is interesting (without having to invent something on purpose). Starting from a programming exercise, I had long ago developed a small utility for ZX Spectrum (and emulators) that I then modified to participate in the "BASIC 10Liner Contest".
The concurrent HIGHLIST program in the SCHAU category, is a utility (to be used mainly as a subroutine) that displays the BASIC program list present in the RAM of a Sinclair ZX Spectrum highlighting BASIC tokens with a certain color (Fig.3).
Basically, it is about accessing the system variables that point to the beginning of the RAM segment of the program in BASIC and then patiently scroll it one byte at a time (in Fig.4 you can find the pseudocoding written in English).

```
9000 REM HIGHLIST (apoalb 2021)
9010 LET ls=1: LET le=9999: LET
s=23635: LET s=256*PEEK (s+1)+PE
EK s: LET e=23627: LET e=256*PEE
K (e+1)+PEEK e: CLS : INK 0: PAP
ER 7: FOR n=s TO e-1
9020 LET a$="": LET q=PEEK (n+1)
+256*PEEK n: LET w=256*PEEK (n+3
)+PEEK (n+2): LET n=n+3
9030 IF q>le THEN FOR n=1 TO 0
9040 IF q>=ls THEN LET a$=STR$ q
: LET a$="    "( TO 4-LEN a$)+a$
9050 FOR m=1 TO w*(a$<>""): LET
z=PEEK (n+m): LET c=3*(z>=165 AN
D z<=194)+2*(z>=195 AND z<=201)+
(z>=202): LET c=c+3*(z=234)
9060 LET k=3*(z>164)+2*(z=14)+(z
=13): FOR g=1 TO k=3: LET a$=a$+
CHR$ (16)+CHR$ (c)+CHR$ (z)+CHR$
 (16)+CHR$ (0): NEXT g: FOR g=1
TO k=2: LET m=m+5: NEXT g
9070 FOR g=1 TO k=1: PRINT a$: N
EXT g: FOR g=1 TO k=0: LET a$=a$
+CHR$ z: NEXT g: NEXT m
9080 LET n=n+w: NEXT n: PAPER 7:
 INK 0: STOP
```

Fig. 3

```
Line  9000     : Remarks

Line  9010     : Initialization of variables ls (line-start), le (line-end).
                 Calculation of s (program start address)  and e
                                  (program end address/variables start address)
                 pointed by Spectrum system variables PROG (addr. LSB-MSB 23635/6)
                 and VARS (addr. LSB-MSB 23627/8).
                 Set foreground and background default colors.
                 Begin main n-loop.

Line  9020     : Initialization of a$, needed to store a line of BASIC program.
                 Each line of BASIC program has the form:
                    - Line number               : 2 bytes (HSB,LSB)
                    - Length text included ENTER : 2 bytes (LSB,HSB)
                    - Text                      : 1 or more bytes
                    - ENTER                     : 1 byte.
                 Calculation of q (line number) and w (length text).
                 Increase of n by 3.

Line  9030     : If q greater than ls then "break;" the n-loop (by means of that "weird" FOR)
                 (the limit value has been exceeded)

Line  9040     : If q greater or equal ls then initialize a$ with q (converted in
                 string and justified right when q < 1000).

Line  9050     : Begin "Parsing Text" m-loop (performed if a$ is not empty).
                 Store in z the content of memory cell addressed by n+m.
                 Choose a color for the BASIC token (black for other characters):
                    - Magenta for functions (coded from 165 to 194)
                    - Red     for binary operators (coded from 195 to 201)
                    - Blue    for commands (coded from 202 to 255)
                    - Green   for REM (coded 234).

Lines 9060-9070: Deploy of a kind of "CASE-WHEN" ("weird" FOR even here):
                     CASE WHEN z>164 THEN add a colored BASIC token (coded z) to a$
                                     (chr 16 represents INK control, chr in variabile c
                                                                    represents a color,
                                      chr 0  represents black, the foreground default color)
                          WHEN z=14  THEN skip next 5 bytes after a numeric literal
                                     ( chr 14 is a delimiter)
                          WHEN z=13  then print a$ on screen
                          ELSE add the character (coded z) to a$
                      END-CASE.
                 End of m-loop.

Line  9080     : Increase of n by w (length of text).
                 End of main n-loop.
                 Restore background and foreground default colors.
                 End of utility.
```

Fig.4

Of course the program can be copied to modify it and improve its performance, sacrificed to remain within the 10-line limit (as per regulation) and to challenge myself not to use "GO TO" (although in the context using the "GO TO" command was allowed).

In [Apo21] you will find the web page where you can freely download HIGHLIST (saved in a file .SNA for ZX Spectrum emulators).

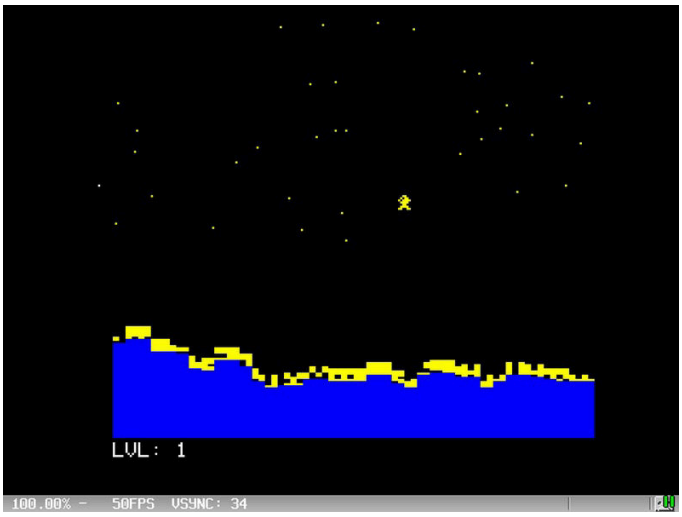


Fig. 5

Winners of the 2021 edition

The results are published in [Kan21c] and the programs can be downloaded from the site indicated in [Kan21d].

In **PUR-80** category he won with 8.36 points "Moon Landing" written by RAX for Oric Atmos (Fig.5).

In **PUR-120** category he won with 8.67 "Noxious" points by Victor "Vitoco" Parada written in Fast BASIC 4.5.2 for Atari 800 XL/XE (or emulators) (Fig.6).

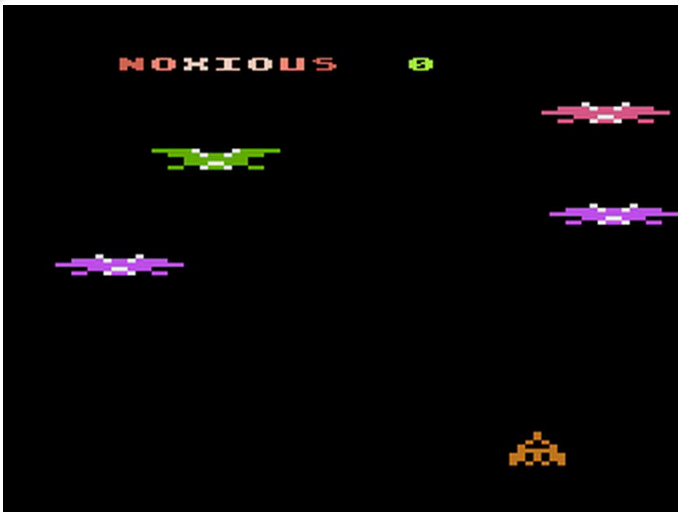


Fig. 6

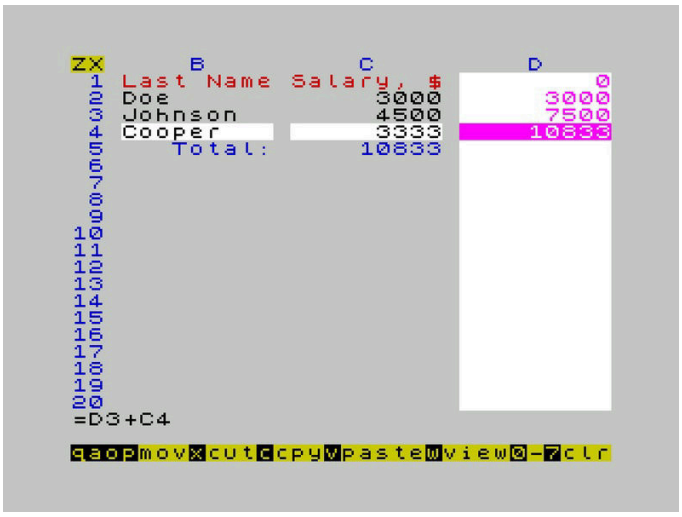


Fig. 7

Fig. 8

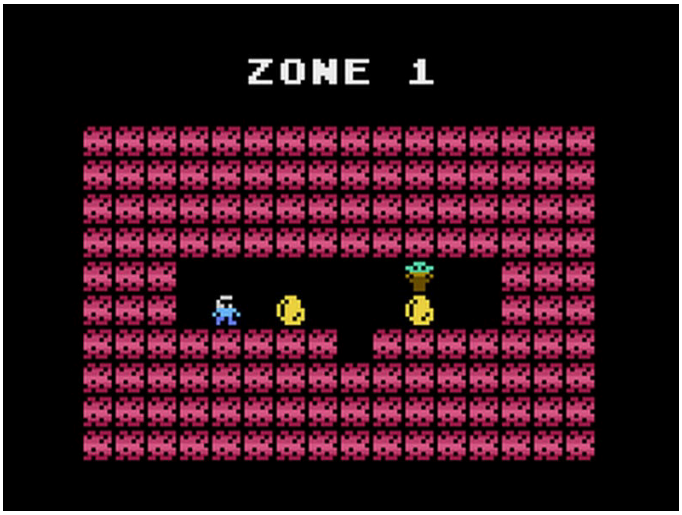


Fig. 9

In category **SCHAU** won with 9.75 points "ZXcel 256" a remarkable version of Excel for ZX Spectrum made by Nick Shcherbyna (Lviv, Ukraine) (Fig.7), known to the Facebook group "BASIC ON THE ZX SPECTRUM".
In **WILD** category he won (no score) "Micro Gauntlet" (Fig. 8) for IvanBasic's ZX Spectrum.
In **EXTREME-256** category he again won 10.17 Victor Parada points with "The Children" (Fig.9) in Fast BASIC 4.5.2 for Atari 800 XL/XE. Also in EXTREME-256, Victor Parada also ranked second with 10.00 points thanks to the Patrol program in FastBASIC 4.4 for Atari 800 XL/XE.
At the time of writing, some special prizes had not yet been awarded.

Comparing the 2020 edition, try to explain what they consist of (apologizing for any inaccuracies):
1) "PUBLIC'S CHOICE" refers to the satisfaction received by a program on the BASIC10Liners Twitter account,
2) "BUNSEN'S SPECIAL AWARD" is named after the site where the programs to be downloaded are located [Kan21d],
3) "FRATZENGEBALLER's SPECIAL AWARD" can be translated into English with "Shoot'em up Special Award",
4. "SKRIEGEL'S SPECIAL AWARD" is the prize awarded by Sascha Kriegel belonging to the Homeputerium group,
5) "YELLOW_MAN'S SPECIAL AWARD" is reserved for programs written for MSX.

Conclusions

Participate in contests only if you already have something ready for the occasion or that can be easily adapted.
Realize the program by putting your heart and soul into it ("spirit" as Steve Jobs said) and, above all, have fun.
If you do this, the program will still be appreciated regardless of the score received and the final ranking.

References

[Apo21] A.Apostolo, "HIGHLIST", https://bunsen.itch.io/highlist

[Bon83] R.Bonelli, "Alla scoperta dello ZX Spectrum", Gruppo Editoriale Jackson, 1983.

[Kan21a] https://gkanold.wixsite.com/homeputerium/home

[Kan21b] https://gkanold.wixsite.com/homeputerium/rules2021

[Kan21c] https://gkanold.wixsite.com/homeputerium/results-2021

[Kan21d] https://bunsen.itch.io/

An introduction to Hollywood - Part 4

by Gianluca Girelli

After some time off the scenes, let's go back to Hollywood language by introducing the use of double buffering. As many of you will already know, there are several animation techniques that are used when dealing with graphic applications, such as video games or technical demos typical of the underground scene. Among them, the most commonly used are: sprites (both hardware and software); layers; double buffering.

Choosing one technique over another depends on many factors, the first of which is what the software has to do. As well described in the Hollywood handbook:

Sprites: Sprites are especially useful when there are not much graphics to be drawn. For example, if you only need to move some blocks or player and enemy sprites around. In this case, it is better to use sprites because Hollywood can refresh the display pretty fast because not much changes.

Double buffering: Using a double buffer Hollywood always needs to refresh the whole display. Although hardware acceleration is used here when possible this can still be quite expensive when you have a 640x480 display which needs to be refreshed 25 times a second. Thus, a double buffer is only recommended when a lot of custom graphics have to be drawn. For instance, the Hollywood examples that draw a real sine scroller use a double buffer because they need to draw a lot of different tiles. Such things would not be possible with sprites because the drawing operations are heavily customized and change every frame.

Layers: Hollywood comes with a powerful layers system which allows you to access every graphics item on the display as its own layer and modify its position, size, and looks on the fly. The layer system is extremely flexible and powerful at the cost of speed so if you need to draw a lot of graphics it might be faster to use double buffering instead.

Here is a recommendation of animation techniques that are suitable for common types of applications:

Board/card games:
Sprites or layers because fast graphics aren't required.

Tetris:
Sprites or layers because there's not much action and screen updates do not have to be very fast.

PacMan:
Sprites or layers. The only thing that moves are the enemies and the player.

2D shooter:
Double buffering because the background is scrolling. Hence, the whole screen has to be updated every frame.

Jump'n'Run:
Double buffer if there is a scrolling background. If the game doesn't scroll then sprites or layers.

Scene demo:
Double buffer by any means. A lot of custom graphics have to be drawn. This is a classical double buffer case.

While the use of layers and the difference between hardware and software sprites will be explored in later tutorials, the scope of this article is to delve into the advantages of double buffering. This animation technique uses two different memory areas (hence the name) to store game graphics (or program in general). The graphic scene is composed entirely in the area called the "back buffer" and, only when the entire rendering cycle has been completed, it is transferred into the graphic memory ("the front buffer") and then displayed on the screen. In

Screen tearing is an artifact that occurs when a single frame displayed on the screen contains information from two or more frames.
This effect occurs when the frame rate of the source (typically a video card) is different from the refresh rate of the display device (e.g. a monitor), so that the buffer is written one or more times with content from other frames.

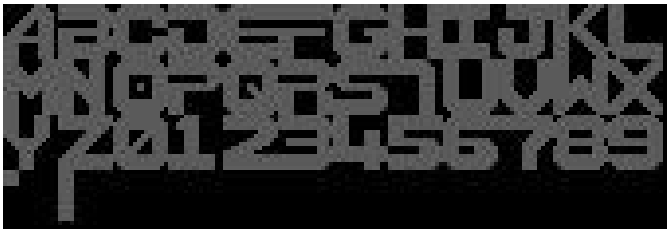
Source: https://it.wikipedia.org/wiki/Screen_tearing

the meantime, the code begins to build the next scene in the back buffer and the cycle continues indefinitely. The inversion between back and front buffer takes place only at the end of the render cycle through a well-defined instruction and this allows to have stable screens, without any flicker or "screen tearing" phenomena (see note).

To better illustrate this technique we decided to use one of the many tutorials accompanying the framework, the "Sinus Scroller": we will thus create an inscription that moves dynamically along a sinusoidal line, while in the background you can see a planet (static) and, behind it, a star field in horizontal movement.

The props we will need are: a picture of a planet (we chose Mars) and a table with alphanumerical characters that

forms the components of our writing. The code's author (Andreas Falkenhahn) chose to compose a table of 12x3 characters (containing the capital letters of the alphabet, the numbers from 0 to 9 and a little punctuation) within which each character has a size of 14x14 pixels.



What happens in the code is this:
- The p_InitText() function initializes the text by comparing the characters that make up the writing to be displayed on the screen (string "TE$") with a reference string ("SCN$").
Then, the x and y coordinates of the letters that make up the writing, with reference to their position within "chars.png", are saved in a vector ("chars[k]");
- A star field (p_InitStarfield() function) is build by creating an array of records whose elements contain: the position (x and y) of the stars, their speed and their color. Different speeds and colors are used to give the impression of depth through a parallax scroll;
- the p_InitWave() function calculates a sinusoidal curve but stores only the y-coordinates. This is done because at this stage we only need to know how the curve develops in height, while the development along the x-axis will be calculated dynamically at each cycle so as to allow it to slide from right to left.

Once the main elements of the code have been defined, we need to write what will be the "Main Loop" and define the rendering cycle. As already documented in the article "Game Coding Notes" (see RMW No. 17 - Italian), this is the center around which our software revolves and should be kept as simple and linear as possible to facilitate maintenance, readability and scalability.

Ideally, the render loop should be contained in a dedicated routine, so you can change or refine it without having to modify all of the main code. This time, however, given the simplicity and purpose of this tutorial, we have inserted the rendering part directly into it. Note the use of the "Flip" statement right at the beginning of the routine: it has the task of immediately pouring the contents of the back buffer into the front buffer without interruptions that would result in tearing or flickering.

The "Flip" statement only works if previously activated with the "BeginDoubleBuffer" statement, as reported in the main code on the following pages. Technically, when you don't need the double buffer any longer, the "EndDoubleBuffer" statement should also be used to terminate double buffering in the current display, but for our purposes this is not necessary.

```
Function p_MainLoop()
        Flip  ; flip front/back buffer
        Cls   ; clear buffer
        /* Render starfield! */
        For Local k = 0 To 60
                Plot(XS[k], YS[k], CL[k])
                XS[k] = XS[k] + SP[k]
                If XS[k] > 330 Then XS[k] = 0
        Next
       DisplayBrush(2, 0, 0)    ; display planet
       /* Update parameters for new x position */
        DIS = DIS - 3
        If DIS = 0
                DIS = 18
                POS = POS + 1
                TEX = Wrap(TEX + 1, 0, TL)
        EndIf
        /* Render scroll text! */
```

```
        For Local k = 0 To 19
                tmp = k + TEX
                If I[tmp] = -1
                        TEX = 0
                        tmp = 0
                EndIf
                DisplayBrushPart(1, chars[I[tmp]].x,
chars[I[tmp]].y, XP[k] + DIS, Y[POS + k], 14, 14)
        Next
        POS = Wrap(POS - 1, 0, 157)
EndFunction
```

Finally, please note the use of the statement "SetInterval(1, p_MainLoop, 1000/25)" in the main code, which serves to time the code at 25 frames per second creating what is called an "event".

Using this statement is not important in itself, but defining the "framerate" ensures that the code runs in the same way on all platforms supported by the framework. In the pictures accompanying the article you can appreciate the Sinus Scroller in action on an emulated Amiga Classic, but I guarantee that it also works the same way on modern systems.

At this point the "magic" is activated by the simple loop (infinite):

```
Repeat
        WaitEvent
Forever
```

which will update our video every 25th second by cyclically recalling the "p_MainLoop" event. The "EscapeQuit(True)" instruction is used to stop the program by pressing the "ESCAPE" key.
Below is the complete program code. See you in the next tutorial.

```
/*********************************************
** Name:        SinusScroller                **
** Author:      Andreas Falkenhahn           **
** Notes:       Music by Heatbeat/Rebels     **
*********************************************/
@VERSION 2,0        /* Hollywood 2.0 is required */
/*
** External data
*/
@BRUSH 1, "Chars.png", {Transparency = #BLACK}
@BRUSH 2, "Mars.jpg", {Transparency = #BLACK}
@MUSIC 1, "Intro.mod"

/*
** set display size to 320x256
*/
@DISPLAY {Width = 320, Height = 256, Sizeable=True,
ScaleMode=#SCALEMODE_AUTO, Mode = "ask", FitScale
= True, KeepProportions = True}

/*
** inits the cosine wave
*/
Function p_InitWave()
        For Local k = 0 To 230
                S! = S! + 0.08
                Y[k] = 120 + Cos(S!) * 101
        Next
EndFunction

/*
** inits the starfield using some different
colors to generate a distance effect
*/
Function p_InitStarfield()

        Local k
        For k = 0 To 9
                XS[k] = Rnd(320)
                YS[k] = Rnd(255)
                SP[k] = 1
                CL[k] = $888888
        Next
        For k = 10 To 20
                XS[k] = Rnd(320)
                YS[k] = Rnd(255)
                SP[k] = 2
                CL[k] = $888888
        Next
        For k = 21 To 30
                XS[k] = Rnd(320)
                YS[k] = Rnd(255)
                SP[k] = 3
                CL[k] = $888888
        Next
        For k = 31 To 40
                XS[k] = Rnd(320)
                YS[k] = Rnd(255)
                SP[k] = 4
                CL[k] = $DDDDDD
        Next
        For k = 41 To 50
                XS[k] = Rnd(320)
                YS[k] = Rnd(255)
                SP[k] = 5
                CL[k] = $DDDDDD
        Next
        For k = 51 To 60
                XS[k] = Rnd(320)
                YS[k] = Rnd(255)
                SP[k] = 6
```

```
            CL[k] = $DDDDDD
        Next
EndFunction
/*
** inits scroll text and brushes + positions for it
*/
Function p_InitText()
        Local x, ypos = 0, 0
       For Local k = 0 To 19 Do XP[k] = k * 18 - 19
        Local SCN$="!
ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789.: "
        Local TE$="                    WELCOME TO
SOME SINUS TEXT DONE WITH HOLLYWOOD. THIS CODE
USES THE NEW " ..
        "DOUBLEBUFFERING FUNCTIONS TO CREATE A
SMOOTHLY ANIMATED DISPLAY. CHECK OUT THE HOLLYWOOD
MANUAL " ..
        "FOR MORE INFORMATION. GREETINGS TO ALL
FELLOW AMIGA
USERS.....................................
........."
        For Local k = 0 To StrLen(TE$) - 1
         I[k] = FindStr(SCN$, MidStr(TE$, k, 1))
        Next
        I[StrLen(TE$)] = -1
       ; init lookup table for character positions
        chars = {}
        For Local k = 1 To 39
            chars[k] = {x = x * 14, y = ypos}
            x = x + 1
            If x = 12
                x = 0
                ypos = ypos + 14
            EndIf
        Next
EndFunction

Function p_MainLoop()
        Flip  ; flip front/back buffer
        Cls   ; clear buffer
        /* Render starfield! */
        For Local k = 0 To 60
            Plot(XS[k], YS[k], CL[k])
            XS[k] = XS[k] + SP[k]
            If XS[k] > 330 Then XS[k] = 0
        Next
       DisplayBrush(2, 0, 0)    ; display planet
        ;update parameter for new x position
        DIS = DIS - 3
        If DIS = 0
            DIS = 18
            POS = POS + 1
            TEX = Wrap(TEX + 1, 0, TL)
        EndIf
        /* Render scroll text! */
        For Local k = 0 To 19
            tmp = k + TEX
            If I[tmp] = -1
                TEX = 0
                tmp = 0
            EndIf
           DisplayBrushPart(1, chars[I[tmp]].x,
chars[I[tmp]].y, XP[k] + DIS, Y[POS + k], 14, 14)
        Next
        POS = Wrap(POS - 1, 0, 157)
EndFunction

;==== Begin Program
Y = {}
XP = {}
I = {}
SP = {}
XS = {}
YS = {}
CL = {}

p_InitStarfield()
p_InitWave()
p_InitText()
DIS = 18
PlayMusic(1)

EscapeQuit(True)
SetInterval(1, p_MainLoop, 1000/25)  ; 25fps
BeginDoubleBuffer

Repeat
        WaitEvent
Forever
;==== End Program
```

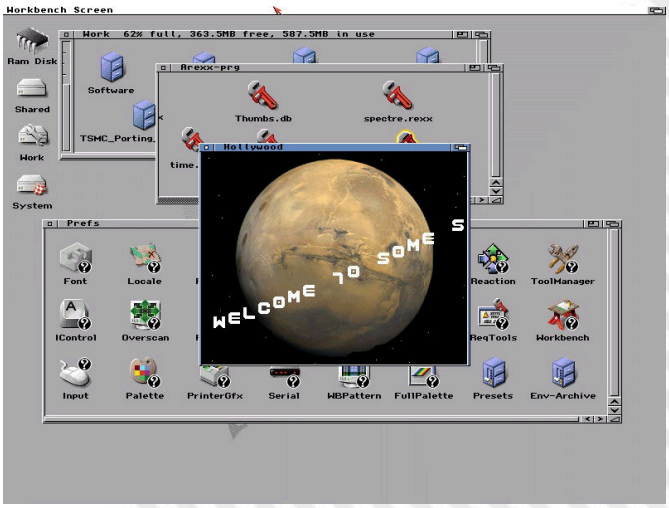


Fig. 2 - Sinus scroller running

Emulators: **DOSBox** ver. 0.74-3

by Francesco Fiorentini

Let's continue our journey into the world of emulators and this time I want to talk to you about DOSBox. This emulator is probably much better known than the Spudace I presented in the last issue, but it is still good to talk about it to introduce the topic and expand our offer of articles to the MS-DOS world.

DOSBox is a CPU emulator and not a simple additional layer, it does not need to be run on x86 machines, thus being independent of the hardware on which it is run. Although once executed it seems to be in front of the dear old DOS, only a limited set of commands have been implemented, but enough to run programs.

Where to find it

DOSBox can be downloaded from:
http://www.dosbox.com/
At the time of writing, the current version is 0.74-3.

On the DOSBox.com site you can also find all the information you need to run, configure and use DOSBox in the best possible way: you can then find a manual:
https://www.dosbox.com/DOSBoxManual.html

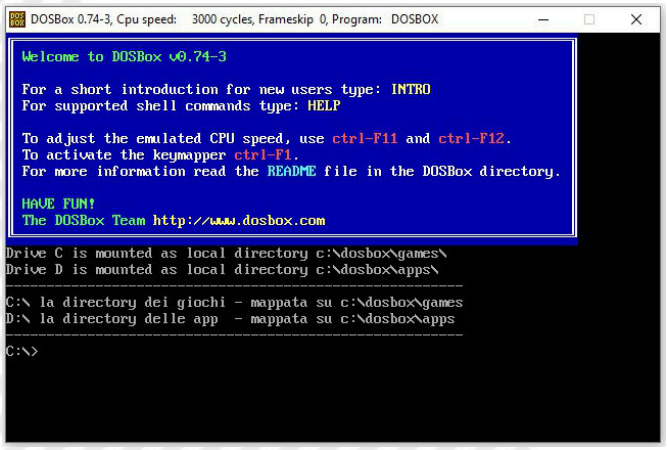

DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DOSBOX
Welcome to DOSBox v0.74-3
For a short introduction for new users type: INTRO
For supported shell commands type: HELP
To adjust the emulated CPU speed, use ctrl-F11 and ctrl-F12.
To activate the keymapper ctrl-F1.
For more information read the README file in the DOSBox directory.
HAVE FUN!
The DOSBox Team http://www.dosbox.com
Drive C is mounted as local directory c:\dosbox\games\
Drive D is mounted as local directory c:\dosbox\apps\
C:\ la directory dei giochi - mappata su c:\dosbox\games
D:\ la directory delle app - mappata su c:\dosbox\apps
C:\>

Fig. 1 - DOSBox started with mouted directories

and a basic tutorial:
https://www.dosbox.com/wiki/Basic_Setup_and_Installation_of_DosBox.

Being born as a gaming application, the main purpose is to be able to run DOS games otherwise impossible to start in current versions of Windows, the site also hosts a list of perfectly supported games.

```
Z:\>dir
Directory of Z:\.
COMMAND  COM                20 01-10-2002 12:34
AUTOEXEC BAT               350 01-10-2002 12:34
KEYB     COM                20 01-10-2002 12:34
IMGMOUNT COM                20 01-10-2002 12:34
BOOT     COM                20 01-10-2002 12:34
INTRO    COM                20 01-10-2002 12:34
RESCAN   COM                20 01-10-2002 12:34
LOADFIX  COM                20 01-10-2002 12:34
MEM      COM                20 01-10-2002 12:34
MOUNT    COM                20 01-10-2002 12:34
MIXER    COM                20 01-10-2002 12:34
CONFIG   COM                20 01-10-2002 12:34
   12 File(s)              570 Bytes.
    0 Dir(s)                 0 Bytes free.
```

Fig. 1 - DOSBox command in directory Z:

Where to start

Once you start DOSBox, you will be faced with a Z:> directory. This is the DOSBox default directory that contains the basic commands (see fig. 2) to run all the subsequent DOS programs that we will install.
The first thing we need to do is create a directory in our real drive, for example C:\DOSGAMES and mount it in DOSBox using THE MOUNT command:
MOUNT c: c:\DOSGAMES

This instructed DOSBox to use our C:\DOSGAMES directory as the root of its C: disk. Obviously all the subdirectories that we will create in DOSGAMES will be displayed by DOSBox and we can use them accordingly.

If you look closely at Fig. 1, you will notice that my DOSBox has already mounted two directories at startup. One dedicated to C: games and one dedicated to D: applications. If you want to do the same, you can copy my simple configuration. The first thing you need to do is locate the DOSBox configuration file. Obviously this is stored in different places depending on your guest OS. Fortunately,

the DOSBox manual helps us:
(Windows) "Start/WinLogo Menu"->"All Programs"->DOSBox-0.74-3->Options
(Linux) ~/.dosbox/dosbox-0.74-3.conf
(MAC OS X) "~/Library/Preferences/DOSBox 0.74-3 Preferences"

Once you find the conf file, open it with an editor and add the following lines to the bottom of the file:

@ECHO OFF
mount c c:\dosbox\games
mount d c:\dosbox\apps
ECHO ----------
ECHO C:\ the games directory - mapped to c:\dosbox\games
ECHO D:\ the app directory - mapped to c:\dosbox\apps
ECHO ----------
c:

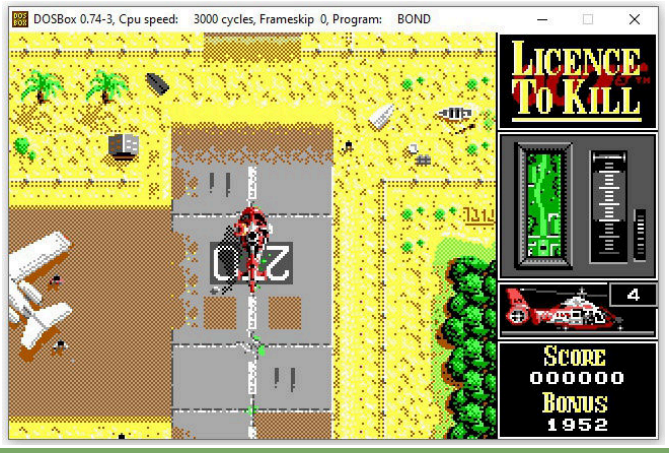
Replace **c:\dosbox\games** and **c:\dosbox\games** with directories of your choice. Restart DOSBox et voilá!
If you check the **AUTOEXEC.BAT** file in the Z: directory, you will notice that these lines are added to this file.

How it works

So far everything is very nice and interesting, but does it work well? Of course it does! I copied the 007 License to Kill game to the c:\dosbox\games\bond directory. Once DOSBox started, I went to the c:\bond directory and ran the bond.com file. The game started at the first shot, just like it was played on a real DOS machine.

After a brief presentation we move on to the real game and this also works perfectly. The function keys are correctly mapped and the sound is present and emulated correctly.

Obviously, you can use DOS applications in the same way.



Technical Features

DOSBox is able to emulate the following hardware.
CPU, almost complete emulation of
- 80286
- 80386
- 80387

Graphic support:
- Text mode
- Hercules
- CGA
- EGA
- VGA
- VESA
- Tandy
- S3 Trio 64

Sound Support:
- Internal speaker/Buzzer
- Creative CMS/Gameblaster
- Tandy 3 voices
- Tandy DAC
- Adlib
- SoundBlaster 16
- SoundBlaster Pro I & II
- SoundBlaster I & II
- Disney Sound Source and Covox Speech Thing
- Gravis Ultrasound
- MPU-401

Additional Hardware:
- DOSBox can emulate a nullmodem serial cable over network and Internet and the TCP/IP and Tuneling IPX protocols.

I copied the DBaseIII PLus directory into the App folder: C:\DOSBOX\APPS\dbase3 and once I started DOSBox I went to the D:\dbase3 disk and launched the dbase.exe file. The DBASE III Plus program ver. 1.1 left immediately doing his duty.

DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DBASE
Set Up Create Update Position Retrieve Organize Modify Tools 06:29:01 pm
Database file
Format for Screen Query
Catalog View
Quit dBASE III PLUS
ASSIST
Move selection bar - ↑↓. Select - ←┘. Leave menu - ↔. Help - F1. Exit - Esc.
Select a database file.

If you've never tried it, do it! A world of opportunity will open up for you. For our part, we will give more and more space to the DOS; it deserves it!

KICK OFF 2: when the football is protagonist

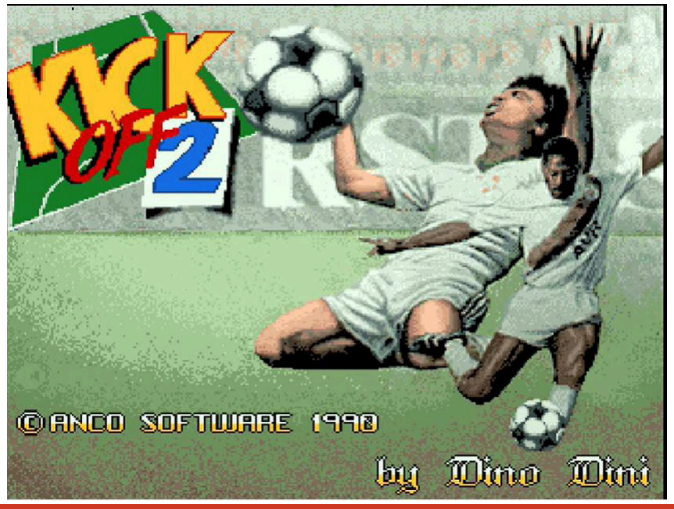
by Mic the Biker Novarina

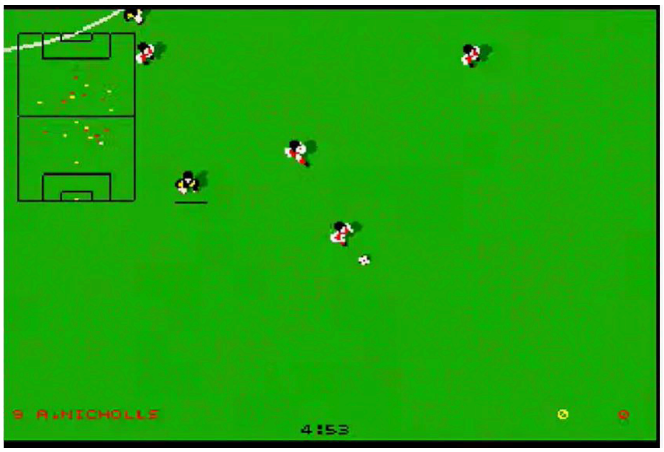
Hello friends of RetroMagazine World, in issue number 6 our brave Edoardo Ullo delighted us with a splendid comparison between the two football giants who dominated the scenes during the '80s and in the first half of the '90s: Sensible Soccer and the Kick Off series. Now it's my turn to pay my personal and proper tribute to Kick Off 2, the football game that, for me, took the computer football game to a higher level. Take a deep breath, tie your boots and keep an eye on that ball, because this time it is the absolute protagonist. Let's start with our trusty DeLorean and set the date back to 1990: in that year a game appeared that was the asteroid decreasing the extinction of a certain type of video game of football: that name is Kick Off. A cataclysm of biblical proportions that gave rise to a new way of interpreting football managed by joysticks and buttons.

Those like me who did all the entire path starting with International Soccer for C64, saw the football games go great, both at home and with the arcades. The good Andrew B. Spencer wrote one of the monuments of the ball in the form of bits, the classic side-view game where the only options were the color of the jerseys and whether to play in one or two. The years passed and as the members of the Band of Borgo San Paolo began to join in company, the first computers began to appear. Obviously among the first games there was that primordial kick, where you could go from side to side of the field with the ball glued to your head. The game of the ball for almost everyone was, at least for a period of life, synonymous with aggregation and its video game transpositions were no less. We met there and we challenged ourselves. It was like a great ritual: we met and each of us pulled his own personal Joystick out of the backpack. It didn't matter if the host had thousands of them scattered around the room, everyone had his own. Epic challenges began, games that are still subject to remembrance today, more than 30 years later. They have become stories, those that were all told in a circle in front of the fire: this is how the oral story becomes myth, it becomes legend.

To understand Kick Off, this brief excursion was necessary, because the football game in question was the end and the beginning of everything and to understand what the second chapter was, it is necessary to explain the first one. Kick Off is the first video game in the series created by Dino Dini in 1989 and published by Anco for Amiga and Atari ST, and obviously also on the eight-bit platforms such as the Breadbin, the Zx Spectrum and console. But who was this unknown author who suddenly came to disrupt the world of video games of that period? Dini is an Anglo-Saxon programmer of clear Italian origins who at the age of 13 created his first video game for the Acorn System 1 microcomputer. It was an eight-bit microcomputer from the late 1970s designed by a woman, a case that I think was more unique than rare, especially in those years. The good Dino continued his journey by creating games that were published by the English publisher Bee Bug. This helped to showcase him so much that in 1989 the video game publisher Anco contacted him. The idea was clear, namely to create a new football video game. This is how Dini realized what he had in mind for a long time, namely a football game that looked very arcade but at the same time realistic. But how could you do something like that? First of all, starting from the bird 's-eye view, the only one that could ensure consistent proportions between the field and the players, and then thinking about the real protagonist of the football game: no player on duty nor the legendary champion of the past but the ball. That's how the legend was born. Not the "big name" international soccer or the "champion" football. Simply Kick Off, kick-off football: no name was more suitable.

Here we leave for another universe, dear readers, here we interpret the game of football as faithfully as possible but in the most unthinkable way. Think about your football experiences for a moment: run on the field and the first thing you do is throw the damn ball half a meter ahead of your foot. This is what the revolution is all about: the ball won't stick comfortably to the feet of our little men.





We will have to chase her, maintain control and fight with the opponents to get to the penalty area. I said at the beginning that here the ball is the real protagonist: you see it moving away from the shoe and it always seems to be too far away. It assumes trajectories totally contrary to the player's positioning.

To master Kick Off it is necessary to deepen the concept of reverse reasoning: you must reason backwards with respect to any football game released until that day. First the lever was used to move the player and then, pressing the button, a step or a throw was made. Here, forget everything: to make a pass you have to keep pressed when the ball is away from the foot. Doing so, once the ball is reached, the player stops, can rotate on himself and, at the time of release, makes the pass. The good Dino Dini had thrown the glove, a challenge that only he was able to take up. Before we go any further, let's go back to school. I play the teacher part and you play the student part. But you, dear readers, are the best students ever, so I already know that my question will be trivial. Let's open the math book on page 10, there is a small problem to solve: Dino has Kick Off and Dini has Player Manager (another of his masterpieces). What happens if the two guys put the two together? Everybody with your hands up! I knew you were the best. You don't need any teachers. The answer is only one: Kick Off 2.

Kick Off 2 maintains the formula of the previous football

chapter and elevates it even further towards simulated football, while maintaining a speed of execution that even the most acclaimed arcade could not guarantee. Within a year Dino Dini had created something that even the arcades envied: he had created THE football game. Kick Off 2 was released in 1990 by Anco for Amiga, Amstrad CPC, Atari ST, Commodore 64, MS-DOS, ZX Spectrum and other platforms.

The game maintains the basic structure of the previous title by introducing many detailed improvements. The view always remains from above, the graphics are slightly more defined and the ball management reaches absolute levels. Here the dribbling method is scientific, even more than in the first episode. As in it, it consists in hooking the ball by pressing the button and then determining the rotation of the joystick to unload the passage, which will be carried out only when it is released. But here precision becomes a weapon, the mental speed required is elevated to yet another power: the gameplay is now even faster and tighter. There is no breath, you fly from one part of the field to the other while our players put team tactics into practice. Yeah, here's the math problem from before. Kick Off 2 reaches the level of legend thanks to the tactical compartment that opened an almost infinite number of scenarios. It was the heart of Player Manager that beat hard, you could feel it, you could touch it with your hand during the game. The leap in quality between the first and second chapters was staggering. You become a strategist, you can adapt the movement of each player according to the strategy applied during the training on the pitch. The world of strategies and tactics opens up: in addition to the basic ones, they can be built ad-hoc by importing them from Player Manager. A madhouse, my friends, I can assure you. Once you have mastered the control system you start to see the basic patterns and there is already enough stuff to make modern football games go directly in the pocket. And since now without a thousand combinations of buttons you don't even have one that is better...

In Kick Off 2 you have to think with at least one step ahead, otherwise you don't even hold the ball. The high view allows a great glimpse and the radar here is really needed, it's not a graphic detail put there because you have to have it. The glance confirmed that, based on our pattern, the player would soon run exactly where in two steps we would place the ball. We of the Band of Borgo San Paolo were totally crazy about this game.
In the backpack together with the Joystick you would surely have found the inevitable floppy with the custom

tactic. Everyone with his own controller, everyone with his own game ideas. Every night we were there, we always ended up playing with Kick Off 2 and spitting in our faces. The ritual was always the same: first a look at the news, then a few sessions on the musical trackers and finally, in silence, just a look to understand each other. The eye of the tiger, the tricky expressions: the matter became rather serious, the challenge began. A few moments before kick-off, just one sentence: "I'll grind you." And it was the beginning of the end.

Kick Off 2 carried other improvements compared to its predecessor: here we will find multiple football fields, equipped with a different graphics and their own characteristics of bounce and inertia of the ball. The jerseys of the teams are customizable, you can play the league, the cup or the classic championship. The number of possible participants simultaneously rises to four, always thanks to the help of door multipliers that for us miserable were a mirage. In hindsight, I think it was a good thing that we could have played in four, two per team, at Kick Off 2. I don't think I'd be here to write to you since they were going to find four dead bodies the next day... The verbal fights between the players were always calmed by those who were waiting for their turn: if everyone had played they would have even found blood on the ceiling. In Kick Off 2 we find a new devilry that made the games even more incredible: it was aftertouch. This option is lethal, allowing to give an effect to the ball once the shot is made. In practice, it was possible to kick into effect by acting on the controls immediately after pulling. This spectacular option turns a trivial shot from inside the area into a possible goal. And what about how direct kicks can be made right over the barrier? It wasn't easy, but taking a roll over the barrier almost always meant scoring a Eurogoal.

In the game we also find wind intensity and salvageable replays to elevate our ego to the highest levels. And for the first time the referee has a precise connotation, it is no longer a mere detail. The black jackets (the referees) have their own specific characteristics: we will have broad-sleeved referees, with whom practically anything can be done. Then there are balanced referees and very strict referees who will whistle foul at every tackle and red tags will be the rule. The Kick Off saga has entered the legend and it is still common practice today to beat an "angalo kick" or take a well-deserved "gaillo card" (very famous translation errors in Kick Off - Note by the translator), because in these cases even the wrong translations assume epic connotations. In that 1990 I signed up for the Kick Off championships but I made little progress, the participants were of a monstrous level and Luca Caldiero won that epic edition. But we of the Gang were still good at playing it, also because we spent hours at the computer. Part of the Legend are the infamous palombella shots struck that quid after the midfield circle, not one pixel before or one after. Dry pull and pull back, there was no escape, the ball would be bagged under the crossbar. And what about the always lethal "stop and goal"? Enter the area, press button, stuffed ball, restart and throw: net ball and empty goalkeeper.

BIKER THOUGHTS

To put it bluntly, it all seems trivial, but only those who have spent a good part of their lives playing with it know that it is not. We were the ones who made it sound banal because watching us play seemed like a football video game like another. No, it wasn't. The ball here was the absolute protagonist, it took all the credit that its importance imposes. I started playing with it again and I am honest, I can no longer do an action worthy of this name, the ball runs away and three touches in a row are no longer in my ropes. But I'll be back, I assure you. It's the magic of the DeLorean, of the time machine: recharging and starting over like never before, to get back to being good. And then I refuse to play any faster than I can see, and other than that, it's just a matter of reflexes. It will have to be years before this series finds a worthy opponent, the SWOS that for many is even better, but not for me. Kick Off 2 is an experience to try, one of those colossal games that makes even modern products pale, equipped with extraordinary graphics and super-realistic animations, but devoid of soul. Especially without that unique protagonist that has characterized this series and that, even today, makes it unique in its kind: the ball.

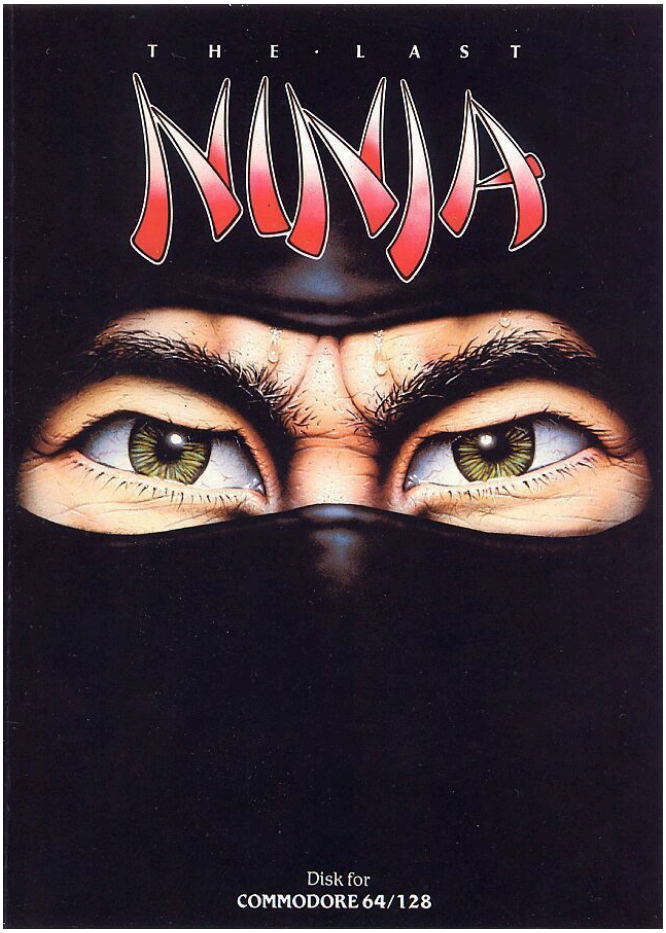
The Last Ninja: legends never die!

by Mic the Biker Novarina

In order to better frame what you are about to read, it is necessary to start from a premise that I believe is important: for me, the game in question is the most beautiful game ever played. I know the second chapter is even better, but the first chapter is like the first love. The reason is always the same, i.e. the “lived” factor. I still remember how I discovered this game: a classmate of mine, who always reads our articles, invited me to his house to do homework. As usual this practice was completed in half an hour because there was a new game to try. He had the C128 and put on this anonymous copied cassette, strictly taken from Americans Games, a historic store located in Via Sacchi in Turin.

The loading screen had already bewitched me: those two eyes staring at me as the colored lines of the loader ran marked my existence forever. At that time the loading times from cassette varied from long to titanic but it was not a problem: we chatted with the friend next door, it was our way of being social. Once the upload was complete, an isometric screen appeared in which we could see the character, at the bottom a power bar and a space for the weapon. On the side is an enemy power bar and space for using and holding. Okay, what do we do? The first half hour we spend understanding that to go straight, in an isometric sense, you have to go diagonally. And beware, this is not trivial: to continue the game is essential. The afternoon was flying between a “Pass the joy to me, keep it yourself, let me try, try yourself” and in the end we knew how to go straight, make flips and kick and punch. I bet a lot of people find it incredible to be able to make such a wide range of movements with just one button, but back in time things worked better.

It was time to go home, but not without getting a copy of the tape in question. The bus ride on the legendary 64 never seemed to end. I just wanted to go home, turn on



my C64 and press play on tape. I remember perfectly well that I had dinner in about 6 minutes at 6.30 p.m. and then slid into the room. It was good times when we weren't weighed down by the long cassette loads: now if we don't have everything ready we immediately lose patience and start (but not me...) criticizing. Long uploads often featured loaders that were authentic masterpieces. The coloured lines were almost hypnotic, shielded by sublime graphics were the protagonists and often accompanied by music of incredible level. Especially those minutes gave us time to think. To what? Well, everyone had their own business, age was teenage, so it could be the classmate we liked, or the thought of the class assignment the next day. A task for which we had studied only half an hour short due to the "fault” of the new video game. Loading finished, we're going! We are on the island of Lì Fen and we will have to guide Armakuni to the palace of shogun Kunitoki, the bad guy on duty: we already know how to move so we start the exploration phase.

Last ninja is a special game, a small big revolution in 1987. It's isometric, there's action, and there's adventure. You have to defeat enemies and on the way you find

weapons: for this reason at the beginning it is essential to go down, avoid the clash in order to run for the sword. Once this is done, the game can really start. With a single button you have three types of attacks: slow or fast hit and a hit from above, the latter the most lethal. Initially you do not immediately grasp how articulated the game is but you soon understand it, when you begin to find strange objects that are not weapons. You have to interact with the keyboard to select weapons and inventory items. Dying at first is definitely easy and only with time you begin to understand the dynamics of the game. The days follow each other and thanks to the tenacity of us players we move on to the next levels. Collaboration was essential, even though there were no cell phones, internet and social networks. But there was the good old phone at home, the one with the wheel, and the line became hot: as soon as you could pass an obstacle, you called your friend to communicate how you had done it. You could spend whole afternoons wandering a level looking for a clue without ever being able to get past it. Sometimes discouragement took over and long phone sessions were held to try to get an idea.

The next day at school, during the interval, socratic discussions were held between players and possibly afternoon study groups were organized, where little was studied but much was fought to try to move forward. After all, we were simple boys, we aimed for average results at school only to be allowed by our parents to play with our videogames! The game turned out to be truly splendid because despite the high difficulty it did not discourage enthusiasts.

Game structure

Lin Fen Island is divided into six parts, each of which is full of puzzles to solve. The sections concerned are: the wastelands, the wilderness, the Palace garden, the dungeon, the Palace, the inner sanctum. Already in the first level we are faced with two recurring aspects of the game: the seemingly impossible parts and the approximate collision of the main sprite with the bottom in the parts where you have to jump precisely. It was initially stressful to see the character sink into some river while standing on a rock! All that remained was to try and find a sequence of jumps without caring about the position of the stones. And what about the thousands of thousands trials of acrobatics to pass the dragon spitting fire and then discovering that he had to be sent to sleep!

Level two

The wilderness sees us enter a rocky landscape where at some points the only way to continue is to move forward with small joystick movements to avoid falling into the void! And I'm not going to tell you how long it takes to figure out how to climb the stone wall first, as well as the millimetric precision required when walking backwards in jerks and then being able to descend without falling! But on this level, I found the first real challenge that made me sweat. I spent days succumbing to the breath of the dragons at the end of the level. I just wanted to remind you that at the time there were no rescues that simplified our lives, and every day the game was loaded it was a redo from the first level. Days to do it again, you would go for attempts, you would try them all and then almost accidentally touch an object that makes us flashing! It's not all trivial, you literally have to fly because the effect doesn't last too much and there are several enemies. Here you learn to use shurikens to kill the bad guys in one stroke and get immune to the dragon's breath!!! Past level, here we are at Palace Gardens.

Level Three

And on this level, I've almost had a nightmare. After finding a quick trick to pass the beautiful river with a waterfall (I put everything down and then only long jumps: even if you ended up in the water you would not sink and proceed quickly) I later found myself faced with a problem: there was a rose to catch but when I tried I always died. A whole day and then turn off the nerve. After a school consultation, a friend asked me, "Did you get the glove?"

Where, how and when? Man, on level two, forgetting an object on the street had stuck me.

Level Four

Here is the dungeon, a real maze infested with lethal giant spiders, ghosts and skeletons. Here I had to organize myself for a moment: paper and pen to make a map of this level because I realized that I was turning in circles. It is easy to go the wrong way, there are few points of reference, the screens are very similar. When you reach the end you climb up to the Palace level, remember the rope please!

Level Five

The level of the palace is definitively more linear and once you enter it you immediately realize that the enemies are very bad and fast. Nothing worrying, but do you think it can be that simple? Of course you don't! Because it took time to figure out how to get past the screen in samurai armor that fires a deadly spear! After touching the pot and taking the next staircase you reach the final level.

Level six

Inner sanctum is tough. The enemies killed, if you go back in the screens, are alive again (characteristic resumed in the second chapter of the saga since the beginning). There are a couple of riddles to move on, the most beautiful of which remains the one to stop the deadly guard dog. After that, we are in the final fight. Kunitoki hits hard but if we put in three-quarters and use the stick in four shots he's dead! The final screen is right here in front of us. But I won't tell you anything!

The ending, seen today, can leave a little bit disappointed, but at the time I lived it with an almost epic feeling of joy. The sun outside was setting and filtering through my bedroom window as I enjoyed the moment.

Biker Reflections

Last ninja was a game that marked an era. Beautiful settings echoed by a simply perfect soundtrack, a real masterpiece. All the music was composed by that genius named Well Daglish, a true 360-degree artist.

Not by chance the songs of the game in question are still the subject of remakes and remixes of all kinds. Time passed by, but often during meetings among friends, we used to load it.

It was like putting on a movie. Three or four people sitting around me, in silent religious, while for the umpteenth time I was preparing to finish the game. The plot was there, the music was there, and the friends were there, while behind us the sun set again.

CASSETTE ONE
SIDE A
(Back up on reverse side)
THE LAST NINJA
C64/128
© 1987 System 3 Software Ltd. All Rights Reserved
UDK
712
100 50 0
Manu-
factured
in the
UK
Distributed by Activision (UK) Ltd.
WARNING: All rights of the producer and the owner of the work reproduced reserved. Unauthorized copying, hiring, lending, public perfonnance, radio or TV broadcasting or diffusion of this cassette prohibited.
SYSTEM 3

Game cassette (image courtesy gamebox64.com)

NEW GAME!!!

PAPRIUM

Year: 2020
Editor: Watermelon
Genre: Beat 'em up
Platform: Sega Megadrive

Eight years is a long time. Endless time. And it was precisely eight long years that employed Watermelon's "brats" to release their PAPRIUM.
Is the wait worth all this time?

Let's start by noting that the game does not work on some Megadrive systems, in particular on "clones". A choice made by the developers.
The game comes to you rich. Excellent packaging with a good manual and a strange condom-shaped packaging that contains the cartridge to be inserted into the slot.

When you turn on the system, you'll find a weird mini-game with a minimalist design that will require you to turn the system off and back on to play the real game. But why is that?

Game Overview
The game is a classic scrolling hit, with three characters that can be selected immediately and three others that can be unlocked later in the game. Each character is different from the other and this makes the game itself a touch of originality.
One of the interesting and unusual aspects of the game is the blue pill (yes, you understand correctly.. "that" blue pill! NdN) that you can swallow during the session to boost your moves, but that could cause some "particular" side effects. Unusual gimmick but a nice touch to distinguish the game from others of the same kind and, I must admit, it works in its absurdity.
There is a "Story" version that winds through paths and that will also affect how the game ends. The oddity in this case is that once we defeat the boss of an area, we can become the boss ourselves and consequently get out of the game. An interesting but perhaps undeveloped dynamic (or maybe yes? These Watermelons are weird. NdN).

But if the idea of being the boss temporarily seems strange to you, Paprium's rescue capabilities are decidedly bizarre. For example, if we upload a save of a certain location, we will find ourselves thrown elsewhere in the layer. Crazy!!! A disconcerting choice, and I wonder again: Why?
It seems like "controlling" the players by the developers. If we think about this, along with the fakeout of the mini-game, the choice to become

boss in place of the boss and the possibility of disabling friendly fire, much of the game design only makes sense if the developers were joking with us players.

The awaiting, the gameplay and... the problems!

I emphasize... Paprium is a visually stunning game and is the best a Sega Megadrive game can have in terms of graphics and animations.

Watermelon stated that this is an 80MEG cartridge and uses the DT128M16VA1LT, also known as Datenmeister, to reach these technical companies. There are also many technical options for a variety of system features, such as Mega Drive handling sound and various visual effects.

Paprium is a game that takes the best of the beat 'em up genre and takes it all the way from this point of view and in terms of gameplay everything seems satisfactory.

There is also an arcade mode that has no branched or unlockable paths.

However, the game also has some rather annoying bugs. The strike detection system is inaccurate and at some points this becomes frustrating.

In terms of longevity, we're in the middle of something. It's not very long, but it's a lot of fun playing with a friend.

Final thoughts

Paprium is an interesting beast. He has some great ideas that all hard-core lovers will love, but he enjoys trolling the player unnecessarily.

Watermelon has some good ideas here, including having a jack for a feature called Megawire that will allow you to upload your scores and download potential DLCs directly to the cartridge. This is something I haven't seen before and it's definitely quite interesting.

However, mismanagement of game crowdfunding and contact with supporters is simply unforgivable.

This combined with the strange design choices makes it a “strange" end product.

There's potential, but I don't think Watermelon handled things as well as he did.

The numerous bugs are a problem and, overall, I think Watermelon could have done more.

After eight years, I expect almost perfection.

by **Carlo N. Del Mar Pirazzini**

OUR FINAL SCORE

» Gameplay 65%

The quirks of this game are many and will make you scream "WHY?".
The story version is particular but enjoyable, excellent the arcade version.
A greater attention to bugs would have raised the final vote for sure.

» Longevity 80%

It's not super long but it's fun to play and very enjoyable with two players. And then there's the blue pill and a thousand other crazy things inside.

NEW GAME!!!

RUNN ‘N’ GUNN

Year: 2020
Developer: Carleton Handley
Genre: Action/platform
Platform: Commodore 64

It's nice this time for productions on our beloved Commodore 64. There are many games that have been released recently, of all kinds. Developed with passion and, above all, with knowledge. A knowledge that was lacking in THE GOLDEN AGE of Home Computers. It was missing for documentation, for practice or, more simply, for "marketability" of the final product. The motto of the times was quantity while now it really seems to be quality.

This Runn 'n' Gunn is really a quality product, perhaps not incisive or original, but well developed and enjoyable. The two main characters are soldiers. They come from different special bodies and, for an experiment, are fused into a single organism and use this transmutation capability to take advantage of their diversity during adventure.

A classic adventure where you have to free the world from the usual alien enemies and to do it best as players we will have the pleasure of “switching” between the protagonists just during the game, using different skills.

Change status. Switch from one particular way of fighting to another. In many video games we have seen this type of development and it has often been used clearly and decisively, so much so that it is essential to change to finish the adventure.
In this game unfortunately the change is not so obvious and decisive and I assure you that we can continue to fight against the opponents even if we remain in the role of one of the two protagonists, never changing.

That's too bad. What was supposed to be a peculiarity proves to be ineffective.
I have to admit, however, that technically speaking, the game is very well done. Clean, neat graphics where everything moves smoothly on screen. The sound compartment is also very nice.

The controls are immediate and you play it with pleasure. It's not very long, but every once in a while you're happy to relive it. In short, a good platform for action, well produced and with a very low price.

You can find it on the website of the its developer at a cost of 3 dollars (peanuts as we say in Bologna, ndN). Here is the site:
https://carletonhandley.itch.io/runn-n-gunn

by **Carlo N. Del Mar Pirazzini**

OUR FINAL SCORE

» Gameplay 75%
Functional in controls and playable. Unfortunately the transformation does not affect the performance within the game...

» Longevity 70%
It lasts a short, very short time... and it's not difficult... however, a game every now and then is gladly done. It costs only 3 dollars.

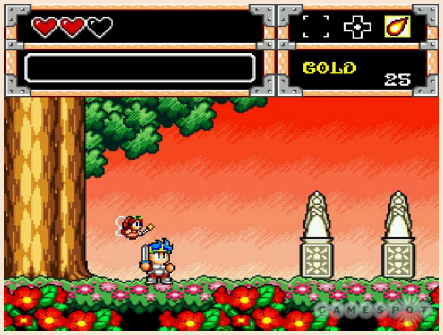
WONDER BOY IN MONSTER WORLD

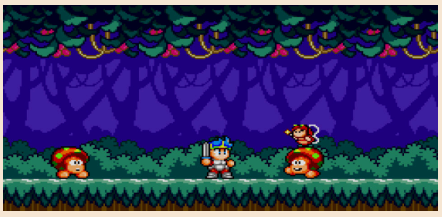
Year: 1991
Editor: Sega
Developer: Westone, Hudson soft
Genre: Platform/Gdr
Platform: Sega Megadrive

Wonder Boy in Monster World is a 2D adventure with some nuances of role-playing and lots of charm.
Known in Japan as Wonder Boy V: Monster World III, it was born as a classic a 2D action adventure originally released in 1991 for Sega Megadrive
In the game, which is the sequel to Dragon's Trap (which we reviewed in the number 24 Italian and 2 Uk), we will assume the role of a warrior named Shion. Your task will be to clear the Monster World of numerous enemies and discover its secrets.
Much of what we will do involves jumping across platforms and attacking enemies with a sword as in any other side-scrolling action game, but the game also has some role-playing nuances, such as the ability to travel between areas freely with the aim of collecting weapons and objects that will make us more powerful.
Nothing to shout at the "MOST INNOVATIVE PRODUCT ON THE PLANET", but with its well-designed levels, with its demanding Boss and with graphics and audio that emanate a sort of "cartoonic" charm this Wonder Boy turns out to be very, very pleasant.

The Monster World consists of six interconnected lands, each containing a village, several dungeons and an end-of-level "Huge" Boss (see accompanying photos to believe, ndN). Shion's Stamina and abilities will increase during the game as traditional in the Wonder Boy saga. We will collect hearts to increase Stamina, we will be able to purchase different swords and armor that will allow us to increase our attack and defense values, and access some spells that have different effects.
Compared to its wonderful and innovative predecessor (Dragon's Trap) we will not be able to transform ourselves into anthropomorphic animals to increase our skills, but to access certain areas we will have to collect some specific objects that will allow us to overcome them. It's a 1991 game and how does it withstand the impact of time?
The technical sector really very well with beautiful graphics even after all these years. The backdrops are just as gorgeous as the animations and character characterization. Also beautiful is the soundtrack that will accompany us in quest.
I remember that in the Italian magazines of the time the game received incredible praise and grades that exceeded 90. Today the game is still very playable and enjoyable but perhaps not as profound as we remembered it as children.
However, it is a pleasant game to play on your Megadrive or in the numerous editions that have come out over the years (it has seen the light from the Nintendo virtual console to the Playstation).

by **Roberto "Il Bardo" Pirazzini**





OUR FINAL SCORE

» Gameplay 85%
You will play with pleasure and it is quite easy to learn. Vary and exploratory.

» Longevity 70%
It's not that complex. It will take you about six hours to complete it.

NEW GAME!!!

SMARTY AND THE NASTY GLUTTONS

Year: 2020
Developer: Eero Tunkelo
Genre: Platform
Platform: Amiga

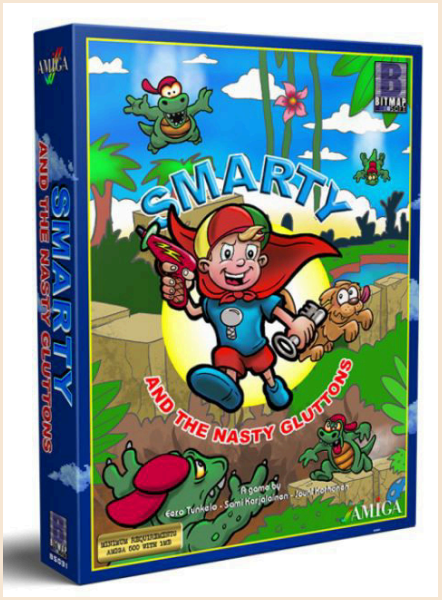
Good old 8-bit and 16-bit platforms are experiencing a second youth these years. Their return to the big market is linked to the phenomenon of retrogaming, which now represents a very solid reality that runs parallel to the world of next gen games. We no longer stop to emulate the old glories of the past, but since some time we have witnessed the increasingly constant release of new video games of very high quality; with quality that in the past we often could only dream of.

There is no exception to this Smarty And The Nasty Gluttons, a fun platform with troubled genesis. History tells us that in 1992 the programming phase began, but in 1995, due to the sharp contraction in Amiga's market share, Avesoft's development was abruptly interrupted. But not all evil comes to harm: with great tenacity and determination Jouni 'Mr. Spiv 'Korhonen and Eero Tunkelo took back the code, unraveled it, improved it and finally finished the game.

As I prepare to upload the ADF file to emulator I searched for the plot they wrote for this platform: the protagonist of the game is Smarty, a schoolboy whose grandfather is a brilliant scientist, in the company of his faithful dog Slurpy. One day Grandpa manages to run his most ambitious invention, the dream machine: he wants to use it for a good intent, that is to help those suffering from sleep disorders. Unfortunately, the cruel Dr. Nightmare, who entered the laboratory, steals the machine to take himself and his grandfather to the dream world to gain control of people's lives. Our little hero Smarty, after falling asleep, is launched into a crazy dream, in which he runs to the aid of his grandfather, accompanied by his faithful dog and the bizarre hat called “Hyno Zapper”: the way to get to the top of the dream begins and will not be a piece of cake!

This is where our adventure begins. The presentation is nice, with little Smarty going to sleep cradled by the litany typical of those music boxes that make a lot of "horror movies of the 80s”. The presentation continues with nice graphics but above all with beautifully swing music, with a Sax that touches perfection. I start the game, dying systematically because I am analyzing and enjoying the beautiful graphics and the spectacular sound: I read that these two components had already been completed in the 1990s so hats off to the work carried out.

Graphically it is really beautiful, chromatically valid, with a good parallax: we are dealing with a platform in which the screen scrolls only to the right and left. The sound is spectacular, the background music is made very well, with clean sounds and beautiful samples. FX sounds are good as well, helping to make the game dynamic optimal. The four Amiga channels are



used properly, the coexistence between music and sound effects is perfect. I start playing, at the beginning of the free spin: it took me a while to understand that falling into the holes of the floor level I emerge from the top of the screen. Enemies abound but we are equipped with a rifle that fires lightning from limited range so we must always be able to eliminate enemies. They are bizarre green beings, they look like a mix between a crocodile and some undefined animal. Once hit they are blocked and remain frozen for a certain time, after which they return active but angrier than ever, also changing colour and aggressiveness.

To proceed on our journey we need to collect keys, to be precise five of them. Getting them is not so trivial, blocked enemies can be pushed off the platforms or through the bottom of the screen to return in the playing area from above. Fortunately, pushing the enemies, I happened to hit one of them spot on an underlying platform and see, in addition to various bonuses, a beautiful golden key arrive. Once you have taken the five keys you can proceed to the next session: each level is divided into five distinct areas. At the end of the last location you enter a rather special bonus stage where you need to collect flying hearts to accumulate extra lives. Here as well we are graphically in front of a good job and also the sound is remarkable. To complete this mission we have a limited amount of time and you have to pay particular attention to the gems fired by the idol placed in the center of the screen: if you are hit, the time will be significantly reduced. Once this beautiful interlude is finished, you will move on to the upper level, with beautiful graphics and a new soundtrack. The game dynamics are always balanced: even when the screen is saturated with enemies, you can always find a way out, as long as you are fast... very fast! From time to time I've been lucky enough to take on spectacular power-ups, like a temporary invincibility that allowed me to slaughter everyone arrogantly and earn a myriad of bonuses.

The game takes us by the hand and entices us to play it. It was a great fortune that this project was not abandoned in oblivion, it would not have deserved to pay the duty of an unfavourable historical period to the sixteen bits of Commodore's house. For some time now the dream has become even more beautiful with the release of a truly rich Deluxe Collectors Edition version: inside an A5 format pack we find a floppy containing the game, an A4 format poster, a set of stickers, the instruction booklet and a distinctive gadget. As a niche genre, retrogaming is once again a very solid and constantly growing reality, with an ever-widening community of players that has returned to appreciate the beauty and the atmosphere that was breathed with this kind of video games, and forgive me if that is not enough.

by **Mic The Biker**

OUR FINAL SCORE

» Gameplay 90%

The controls are simple, in the tradition of "one-button" games, so you'll quickly become familiar with them. It takes a bit of practice to understand the dynamics of bonuses but this is the beauty of the platforms. To be discovered during your play!

» Longevity 93%

Once you get the hang of it, it's damn fun, free of bugs and slowdowns. The difficulty is well calibrated and grows with the right pace. In general, with practice, you can manage to master even the hottest and most critical situations of the game. A challenge that will keep you busy without stressing your nervous system!

CHRONO TRIGGER

Year: 1995
Editor: Square
Developer: Square
Genre: RPG
Platform: Super Nintendo (1995) – Digital channel Switch, Nintendo DS and PC
Version: Super Nintendo

One of the greatest role-playing experiences ever created

Immediately after its release on Super Nintendo in 1995, Chrono Trigger quickly became one of the most unique and impressive role-playing games of its time.

Not only was it developed by the Dream Team of video game developers (Hironobu Sakaguchi creator of the Final Fantasy saga, Yuji Hori creator of the Dragon Quest saga, Akira Toriyama as designer of the characters creator of the Dragon Ball saga, Yasunori Matsuda and Nobuo Uematsu fathers of Final Fantasy to music and Masao Kato of Final Fantasy VII) but it turned out to the mid-1990s gamer crowd as a refined product like few other titles of the time.

Released towards the end of the console life cycle, it has become one of the most beloved role-playing games for this system and is still remembered today as a timeless unique success.

All in a 32 megabit cartridge that was supported by an additional Ram memory powered by a dedicated battery dedicated to saving.

During the development (rather long and created in the middle between Japan and the USA) the game presented an even more articulated and infinitely longer plot, but the limits given by the Snes, by the exit times and some inconveniences of the route, brought some cuts. This translated into an average duration that was not very high, but at the same time gave rise to many under this and to 13 different endings due to the choices you will make during the game.

The protagonist of the game is Crono who together with his adventure companions (up to 7 characters in the party) will have the task of saving his world from the evil alien Lavos.

Some classic RPG elements such as the evolution of parameters, armaments and magic contrasts with a different style of combat that, while maintaining turns at the Final Fantasy, will allow us to make the most of the positioning of our heroes on the battlefield so as to create more powerful blows and combos that will involve even 2 elements of the party in the attack.

Time travel also plays an important role in the game and makes the narrative flow engaging and unique.

Graphically, Chrono Trigger is one of

the most beautiful games ever made for the 16-bit console. Not only does it showcase some of the most beautiful Mode-7 effects seen on the system, but it also has a level of color and detail that are not present in any other SNES role play. Every single area and period of time has its own distinctive appearance and really gives the game the feeling of being even bigger than it already is.

Impressive details abound in every single area; even the characters offer some extremely remarkable animations that make them much more expressive than the standard for 16-bit titles. Pure Pixel Art!

Square has always had a talent for bringing some of the best soundtracks into their role-playing games, but they've really outdone themselves with the soundtrack in Chrono Trigger. He has one of the most beautiful music you've ever heard in a video game. From softer ballads to battle melodies to rhythm, there's so much acoustic intensity throughout the game.

Each piece composed for each area brings a distinctive mood that always seems to adapt perfectly to the scene. In truth, a mediocre track is not included and music keeps getting better and better as you progress through the game.

Chrono Trigger is one of the unforgettable experiences you've never seen on a console.

Together with Legend of Zelda for Snes, it represents the highest point of the genre for Snes and beyond. Chrono has made history, the subsequent Gdrs for the Sony and Dreamcast consoles have resumed many game mechanics and dynamics.

Retrieve it if you have a Snes or digital version on Switch, or if you want you can try the HD version for Nintendo DS and the 2018 port for PC.
In every version it will take you for a walk through time on a fantastic adventure.
Bard's word.

by **Roberto "il Bardo" Pirazzini**

OUR FINAL SCORE

» Gameplay 95%
Simple and intuitive control system with an excellent tutorial that will let you learn how to move through the game worlds.

» Longevity 90%
The basic story is short, but the numerous quests and 13 different endings will keep you glued to the console for quite some time.

NEW GAME!!!

METAMORPHOSIS

Year: 2021
Developer: Kees van Oss and Leonardo Vettori.
Genre: Platform
Platform: ZX Spectrum

What a great game! A real gem these days. A small gem in the Spectrum game collection.

Metamorphosis is a macabre, bloody, difficult experience that will keep you held to the screen.

First of all. I am almost a rookie on Spectrum. Back in the day, I had a C64 and an Amstrad CPC. I've never tried Sinclair's 8-bit computer physically. Just tested some emulated games.

Let's just say I start as a novice without prejudice of any kind. No expectations.
As if turning on the small British home computer for the first time to test the game wonderfully.

How does Metamorphosis look to us?
The purpose of the game is to survive, evolve, eat the enemy and get out of the claustrophobic underground labyrinth where we are.

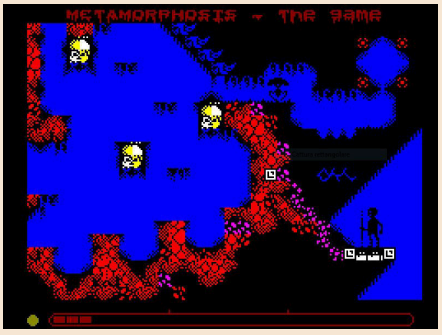
We're not the good guys, we're not the bad guys... we're part of the hive.
We'll have to fight our opponents by eating them. Once eaten we will evolve into a higher, more resilient and more lifelike form.

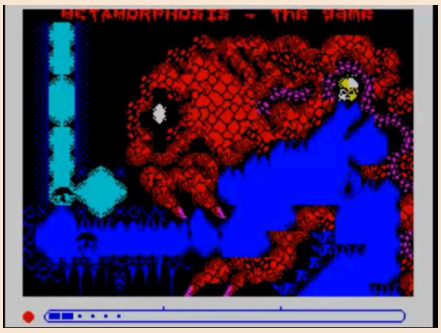
This will allow us to explore labyrinth caves and solve all puzzles.
In order to get out of the cave we have to collect three round icons with our character drawn and take them to the central room with the circles. There's a position in this room to drop objects into the circles. When all 3 are in the circles, a teleport will appear for the next level.

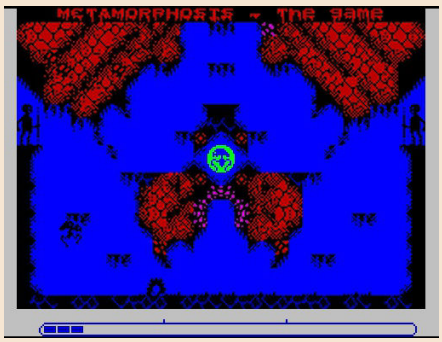
It sounds simple in words, but I assure you that the degree of defiance of the opponents really puts us to the test.

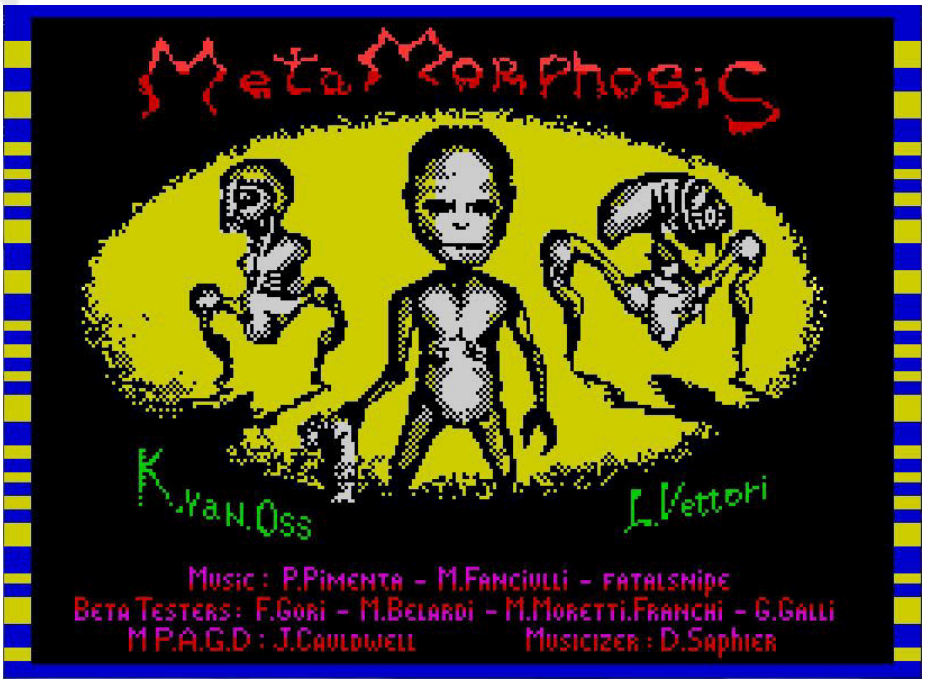
A refined product.
Well developed and animated and with minimalist graphics but well animated and very gore. I love the alien protagonist of the game and his animations.
Sound base and effects suitable for the theme.

You can play with the keyboard or with the help of joypads. The character moves, jumps and spits out his acid with the simple directional cross and fire button.









Simple, immediate and fun.

I found the levels well structured and with a degree of challenge never too punitive but that does not allow carelessness.
A long-lasting and enjoyable game you should try.

Not having a Spectrum "real hardware" I tested the game with the Speccy 5.4 emulator and it ran properly. And if you have 128KB of RAM you can listen to a really excellent music, so a big praise to the audio team that has been able to give body to claustrophobic and disturbing tunes. Well done!

Download it and support the development team with a small donation, it's worth it.
You can find the game here:
https://ennl-team.itch.io/metamorphosis?fbclid=IwAR0rRcn-sMC0-4KxNPZQWkIyNj6ZV6GZZskzO0ppDLxBBRjRfBvgGON8R1E

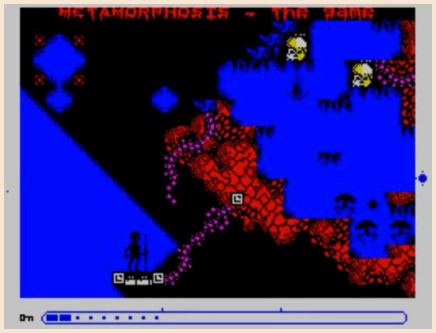
Have fun!

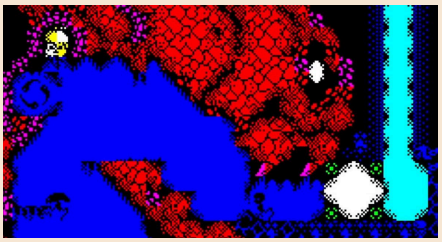
by **Carlo N. Del Mar Pirazzini**

OUR FINAL SCORE

» Gameplay 90%
Simple in commands but well structured in levels and "puzzles". Excellent overall playability thanks to a compelling and a macabre gameplay!

» Longevity 85%
It doesn't forgive mistakes and distractions... and rightly so, but its degree of difficulty will keep you glued to the screen.

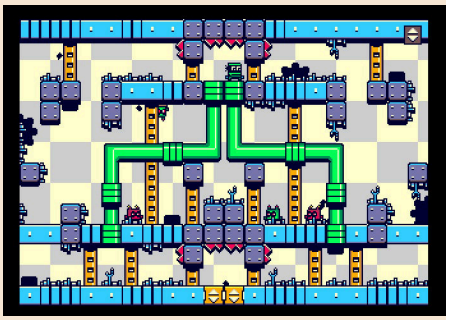


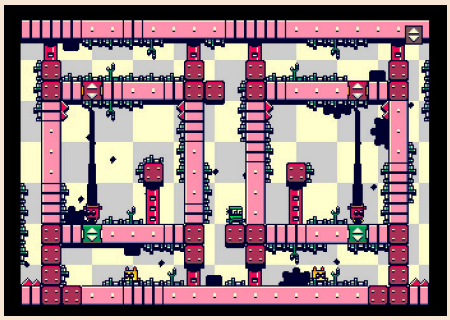


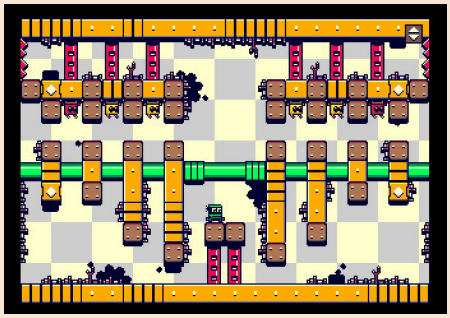
NEW GAME!!!

GRAVIBOTS

Year: 2021
Developer: Retrosouls
Genre: Puzzle/Platform
Platform: Sega Megadrive
Version: Sega Megadrive



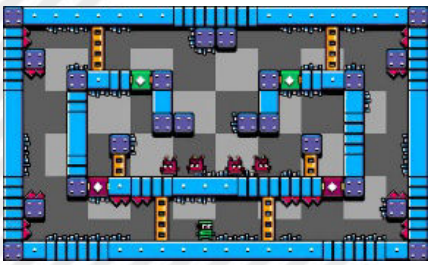




"The purpose of the experiments in the laboratory chambers was to study the nature of gravitational fields.
As a result of numerous experiments, a universal bot has been created that can easily change the local gravitational field.

After closing the research program, we now have a problem: you need to dispose of old robots roaming in the experimental room.

Your job is to drive the latest model of the gravibot, clearing all the old robot rooms using abandoned and forgotten gravitational traps."



This is what the Retrosouls guys tell us on their website and it is the story we will face playing this Gravibots, a colorful homebrew totally free for Sega Megadrive.

The purpose of the game is to eliminate enemies by positioning gravitational traps with which we can modify their path of movement, making it easier for us.

Put this way it seems simple, but it takes a lot of practice to be able to overcome the levels (and different patience! ed. Nith).

Graphically colorful and animated really well, the game really lets itself be played with pleasure and presents a degree of gradual challenge really well done.

Mention should be made of the very arcade soundtrack that recalls Dizzy's games.

The game runs well on emulator or real hardware if equipped with everdrive cartridge or similar.

A pleasant game therefore that I recommend. You can find it here: https://retrosouls.itch.io/gravibots16bit

by **Carlo N. Del Mar Pirazzini**

OUR FINAL SCORE

» Gameplay 80%
Simple and fun. Well developed puzzles.

» Longevity 75%
Difficult but never exhausting. A game that will keep you glued to the joypad.

NEW GAME!!!

MONSTRO GIGANTO

Year: 2021
Editor: RGCD
Developer: Pirates of zanzibar
Genre: Beat em up
Platform: Commodore 64

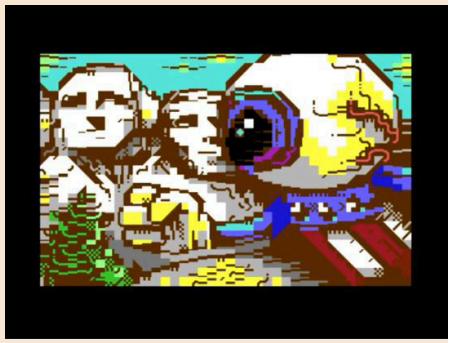
Long live the pirates! Long live the pirates of Zanzabar! W Mojo, JoJo, Gorgo and EyeEye!
Giant Monster is the beat em up you never expected on C64 in the glorious 80/90's.
Masterfully programmed by that genius Antonia Savona, by Lobo and with the fantastic music and audio digitizations by Aldo Chiummo.
It is a classic fixed screen beat' em up with 5 monsters to select (4 already available and 1 to unlock), each with its own skills and moves (fists, heads, bites).
The graphics are entirely made of PETSCII, a variant of the ASCII encoding system used in Commodore computers since 1977, and move incredibly. Super animated, super fluid and really very fun to watch.
Characters are really well characterized, with a custom set of moves, a personal sound intro and graphics, and their own AI (which works really well).

And what about the sound compartment. Amazing digitized audio with a speech that will do wonders on the c64 SID and a perfect soundtrack.
A fun game in double, fun in single and with the Origin tales, functionality that can be unlocked with the story narrated in the voice of each monster.
The combat system with the implementation of fatigue during combat is very interesting.
The game was released in a physical version with incredible packaging.

The pack contains the C64 cartridge, an 18-page manual with the origins comic book inside, a set of four pins, a postcard and an adhesive. All for £35 (you can buy it here: https://rgcd.bigcartel.com/product/monstro-giganto-commodore-64).
If you don't have an "original" C64

the game is also available in digital version and works on easyflash or similar (Ef version) and in digital version for emulators or TheC64 (and you can find it here: https://pirates-of-zanzibar.itch.io/monstro-giganto).
What else to say... don't miss it.

by **Takahiro Yoshioka** and **Carlo N. Del Mar Pirazzini**



OUR FINAL SCORE

» Gameplay 90%
Fast and fun control system, custom move set and fun implementation of fatigue during combat.

» Longevity 90%
Fun in single, fun in doubles.... The Origin Tales is fun too!. You'll play it again and again.

NEW GAME!!!

NEPTUNE LANDER ELITE

Year: 2021
Editor: Phoenix Ware
Developer: C64MARK
Music: Phaze101
Genre: Spaziale
Platform: Commodore 64

Come here, Doc. Doc, come here! More to the right. No, I don't give a s**t! To the right!

And of course you're going to crash, terribly, but it's going to be the greatest crash of your life, believe me. And do you know why you're crashing into that rock on the planet of Neptune?

Because Neptune has never taught you to drive well in the absence of gravity, GOATS! (Neptune, in italian Nettuno, sounds like nobody, nessuno... - Note by the translator). Okay, if you've survived this beautiful, hilarious joke that opens the long-awaited review of award-winning Neptune Lander Elite and you haven't thought about waiting for us outside the newsroom to beat the crap out of us, you deserve to be able to get into the floating world of the latest creature of programmer C64 Mark and his coder-concubine (in the sense that they develop games together) Phaze101, which has gone wild with music and special effects.

With Neptune Lander Elite you will venture on forty screens, choosing how to torture your spaceships: depending on the chosen spaceship (easy, normal, hard) you will find yourself piloting an increasingly stubborn vehicle, less and less manageable and less inclined to slip with sinuous movement into the shells of a planet that has no intention of making landing easy.

And you may wonder, why the hell do you have to land on a planet so unwilling to have an open relationship with your rudder? Soon said. A little because you're GOATS, a little because you're officially looking for surviving coworkers.

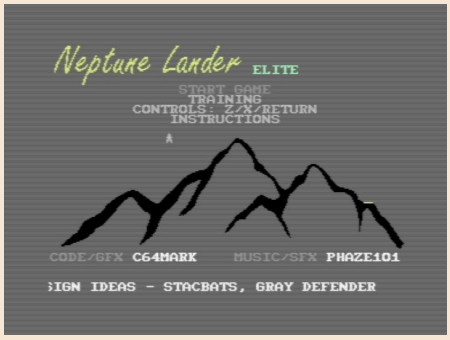
So, give yourselves a hell of a restraint... inside your belly, outside your chest and look up, if nothing else because you have to observe and calculate (quickly) where you want to land.
In fact, for each of the 40 screens, you will be given an average of three opportunities.

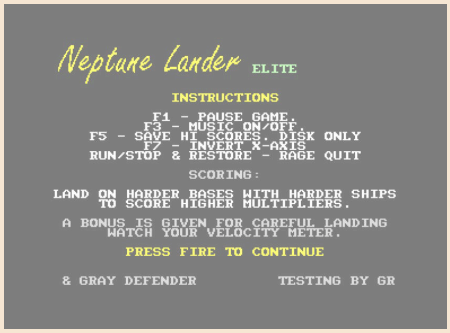
The more you dare the dangerous landing, the less reachable the platform, the more you multiply the points.

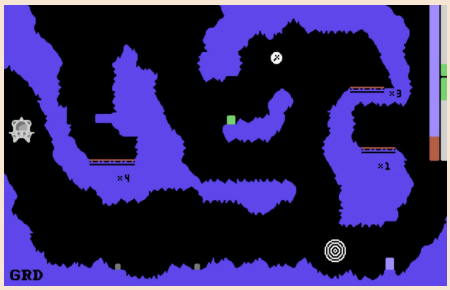
Yeah, but what points? Right question, Lieutenant Colonel Jhonny Birillo, you may sit down, please. Basically the points are calculated on how much gasoline (yes the one that made Jovanotti (...an italian singer) turn his head) you will have left in the tanks when you put the legs of your vehicle on the ground.

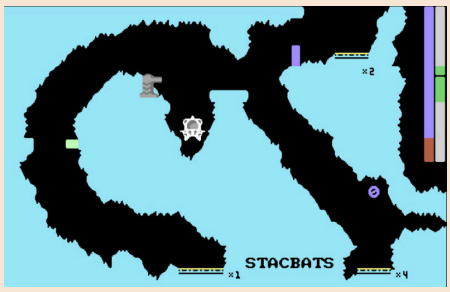
So, to sum up, to make a good score you will not only have to dare to enter deep places that would blush any system of planets, but you will also have to do it in the shortest time possible (but speed is your specialty, right?). Lieutenant Colonel Jhonny Birillo, I know it sounds like a suicide mission, and it is, but I haven't told you everything.

Basically together with caves, passageways, ever narrower slits, you will also have to deal with areas where gravity will drag you to the ground without being able to do anything about it (you will recognize them by two luminous pixels), concluding this deadly list with lasers that want to saw you in two and doors




Neptune Lander ELITE
INSTRUCTIONS
F1 - PAUSE GAME.
F3 - MUSIC ON/OFF.
F5 - SAVE HI SCORES. DISK ONLY
F7 - INVERT X-AXIS
RUN/STOP & RESTORE - RAGE QUIT
SCORING:
LAND ON HARDER BASES WITH HARDER SHIPS TO SCORE HIGHER MULTIPLIERS.
A BONUS IS GIVEN FOR CAREFUL LANDING WATCH YOUR VELOCITY METER.
PRESS FIRE TO CONTINUE
& GRAY DEFENDER
TESTING BY GR





to pass through that always snap when you do not expect it.

Well, now that I've told you the mission, dear Lieutenant Colonel Jhonny Birillo, I'm also responding to the easiest of his objections. Yes Neptune Lander Elite is nothing more than a living tribute to two titles who have made the history of video game entertainment. Or at least one of them.

We are talking about the classic Lunar Lander arcade (1979, Atari) and its "clone" Jupiter Lander (1981, Commodore 64). The mission of paying homage (if not exceeding) these two titles has been fully successful, so much that this title has brought home the prestigious FREEZE64 GAME Of THE YEAR 2020, the annual prize of the English fanzine/paper magazine FREEZE64 and also other “competing” newspapers in UK are cuddling it, acknowledging, a little late, all the success it deserves.

Yes because we have in front of us a really fun, potentially eternal game that puts the emphasis on what all of us (or at least me) look for in a game for Commodore 64 (but not only), that is, the immediate fun, the challenge of beating the best score, in short, the dear old thrill from the arcade (excluding the thrill of having the classic dealer near who was trying to sell you the stuff).

In addition, dear Lieutenant Colonel Jhonny Birillo, I confess that this Neptune Lander Elite stripped me so much that I actually started playing with it and never stopped, improving daily and becoming, with 274,648 points, 17 extra lives embedded in hard mode, the world champion of this lovely title.

What...? Lieutenant Colonel Jhonny Birillo, you say I'm world champion just because no one plays with it and so, having no competition, do I like to win easily?

Oh, that's good. Then let's see what you can do! Neptune Lander Elite is released free of charge on the website c64mark.itch.io (any extra donation will go to the "Centre for Computing History" in Cambridge, UK) or can be purchased in a "physical" version, in cassette format, which can be ordered for a few euros on the website: www.phoenixware.co.uk

Take a copy as long as you can. You won't regret it.

And have a good landing!

by **Francesco “iononsoleggere/ Icannotread” Bizzini**

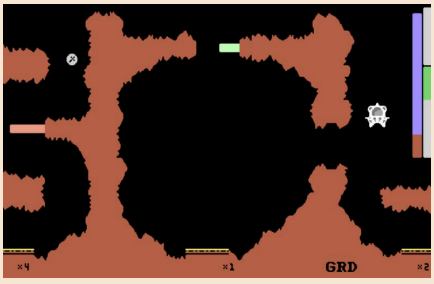
OUR FINAL SCORE

» Gameplay 99%

Absolute. The dynamics of control of the ships are accurate. If you then add that there is also the section of "training" where you can experience all kinds of landing and all the neptunian misfortunes... well, more than that, I think, from the technical point of view, it is difficult to get. A gem.

» Longevity 90%

Forty screens and three types of ships to pilot (with different handling characteristics) ensure a really high longevity. You'll find yourself putting your hard-earned hands on it again and again. In fact, you'll start to think that forty levels are not enough. You'll play it often. A space drug.

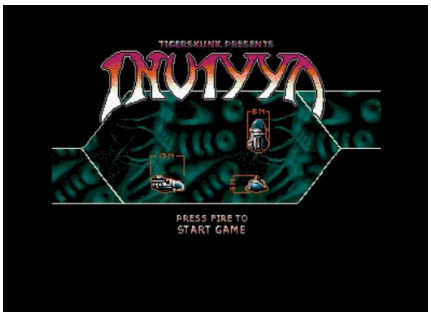


NEW GAME!!!

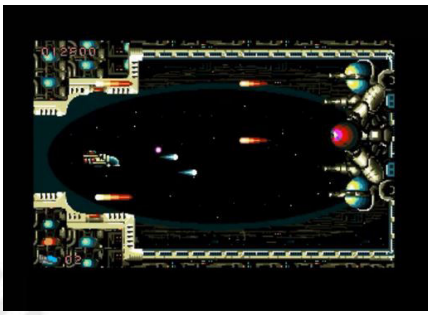
INVIYYA

Year: 2021
Editor: Poly.play
Publisher: Tigerskunk
Genre: Shoot em up
Platform: Amiga

Over the years, the Commodore system has given us a tide of shooters of all kinds and genres.
Some absolute masterpieces of the genre such as Apydia or Banshee, others beautiful but with serious gameplay gaps such as Project X or Agony (beautiful but very difficult).



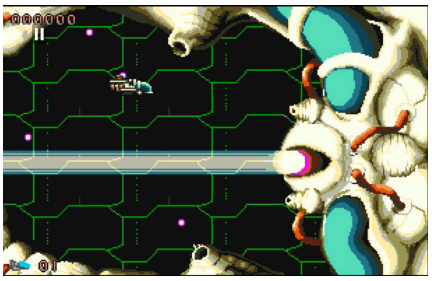
Inviyya is the classic horizontal SHMUP featuring 6 levels each with their enemies and bosses, weapon upgrades, 3-level parallax scrolling and great music and sfx.



It is a game created to be played to the best of all existing Friends, in fact 512k and any other form of 512 kb of extra Ram (slow or fast) are enough. Personally I recommend it on cars with a slight acceleration, but also on the A500 it runs more than decently. The game is very playable and with a good balance of difficulty.

Graphically it was made very well. Nice parallax and nice animations. Everything moves smoothly.
Sound suitable for the genre.

That being said, we are dealing with a good game but not an absolute masterpiece of this kind.



A fun title that will put you through a few hours of play but will not make you scream at the absolute miracle.

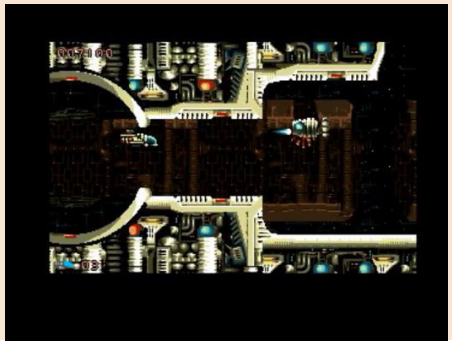
You can download it here:
https://tigerskunk.itch.io/inviyya

by **Carlo N. Del Mar Pirazzini**

OUR FINAL SCORE

» Gameplay 75%
We are not facing a revolutionary game even if all the elements seen in recent years are present. But it can be played pleasantly and has a good difficulty level.

» Longevity 70%
The six game levels will keep you busy for a while then you risk leaving it inside your hard disk for quite some time before playing it again.



XAIN'D SLEENA

Year: 1986
Developer: Technos
Genre: Sparatutto/Platform
Platform: Arcade
Version: Arcade

A really original title I found in the arcades was Xain'D Sleena, also known as Solar Warrior. Its peculiarity lies in the fact that it presents itself as a normal platform, but, between one level and another, it turns into a classic space shooter.

But let's be orderly.

Published by Technos back in 1986, in this title we play the role of space soldier Xain, who has the task of freeing five planets from the forces of evil.

At the start of the game the map of this unknown galaxy appears immediately and we can choose the order of the planets to set free. Each has its own peculiarity: there is the planet of the wild forest, the one submerged by water, the one with a lunar landscape, the one that looks like a desert and the one covered by the lava of a volcano.

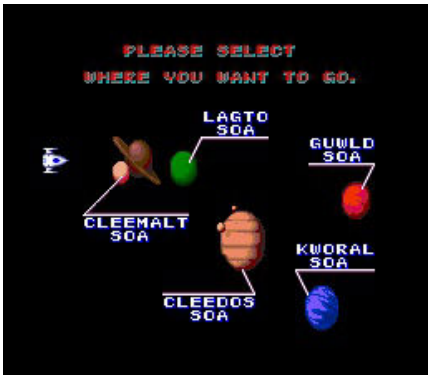
We have at our disposal a red armor and a laser rifle that we can enhance if we collect certain items left by some enemies.

Once the planets are also set free from the fearsome end-of-level bosses, our hero will plant a bomb and we only have sixty seconds to get on the spaceship and leave the planet before it explodes.

At this point the developers surprise us because, before we reach the next planet, we have to pilot the spaceship like a horizontal shooter and see it with asteroids and enemy ships.

But surprises don't end there. In fact, once the fifth planet is destroyed, a sixth planet will appear on the map, the seat of the enemy fortress.

In this stage, before we trigger the last bomb, we have to confront all the bosses we met on previous levels.



The sprites are colorful and detailed with really beautiful backdrops, while the enemies are varied and difficult, ranging from classic aliens to even prehistoric dinosaurs.

The sound section is also noteworthy with different tracks for each level and musical effects suitable for each phase of the game.

This game has also been converted for several consoles such as the Atari ST, Commodore 64 and Amiga.

In conclusion, I really recommend you to rediscover Xain'D Sleena, you will find a little Contra and a little Turrican in it, in addition to its many characteristics that make it original even today.

by **Querino Ialongo**

OUR FINAL SCORE

» Gameplay 90%

In Xain'D Sleena we have a button to jump and one to shoot. But this classic game system is enriched by many ingenious gimmicks that make it very enjoyable even today.

» Longevity 90%

Like all arcades, Xain'D Sleena can be finished in about 40 minutes. The initial possibility of choosing the path, however, makes it varied and innovative.

PUB TRIVIA SIMULATOR

Year: 1989
Editor: CodeMaster
Genre: Trivia
Platform: Commodore 64

We are already at the second Easter we spend closed at home, with prohibitions and restrictions; bars, restaurants, theatres and much more are all inaccessible. The time to play and discover new games, present and past, has never been and will never be restricted in any way.

During an afternoon like any other, I decided to have a dart game, with the marksmanship I have in my room and between one center and another I thought back to all the pastimes and entertainments that could be found in bars and pubs. The shooting was one of those, although during a video game I had read the phrase “drinking and darts do not mix”.

Fate wanted me to discover a game, which probably would get along with a beer, it's a quiz game... That's right! Just a game of quizzes like early evening tv games show that our grandmothers don't miss an episode.

Pub Trivia Simulator takes place on the table of a pub, but without the nice and sly conductor as on TV, with the possibility of choosing a series of cheerful challenging characters, of both sexes.

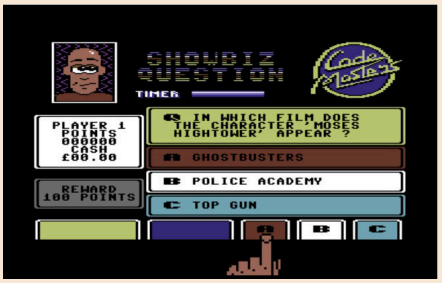
We will have at our disposal a pyramid depicted in blocks, each with the kind of question we decide to choose (sports, cinema, etc...) and in addition there will also be wildcards that will make us gain the next turn without answering any questions. Purpose of the game? Reach the peak! To do this you will have to study English since the game is entirely in English and above all read and learn as much as possible; the questions will not be as easy as the one that asks us the name of the Real Madrid stadium.

What about gameplay? Playability cannot be defined when only three keys are available to answer the question, but all this is compensated by longevity. You will not give it up until you have won at least one game or filled up with information contained in the questions. If you miss the first, you will have another chance, while from the second onwards you will have to think twice before pressing the button, otherwise no gold coins! Oh, no, sorry, this is not Jeopardy. But at least one drink is allowed, come on!

As said, the game is entirely in English and I think a basic level is enough to understand the questions, it is not necessary to be a mother tongue (fortunately), but at least of the smallest fans of sports, cinema and music that are not so unknown subjects.

While I am writing this review, Easter is almost over and I recommend you play this game at the next party and possibly in company; it will be great fun and a great challenge with your friends and family. Maybe even giving you a few small drinks as a reward for answering all the questions correctly.

by **Daniele Brahimi**



OUR FINAL SCORE

» Gameplay 30%
Playability: options a,b,c?

» Longevity 90%
You won't come off easily, not before winning at least one game but, no prizes or gold tokens!

Mixed feelings

Recently on Facebook and other social networks, I have been noticing an incredible increase in posts related to retrocomputing. Articles in newspapers and magazines (also in tose not IT-oriented). I even watched a documentary on Netflix about retrocomputing (for the sake of the record, it received very divergent opinions...). Never before has this subject been so mainstream.

- That's great! I mean, good!
- Hhmm... Well, maybe...just OK...

- But, hey? Aren't you happy?
- Of course I'm happy!
- In fact... No… Er... Actually, I don't know.

Okay, enough of this pantomime. As you may have guessed from the title, I'm overwhelmed by a series of conflicting feelings. I am glad that our common passion has finally acquired some popularity and that more and more people are interested in it, but as it happens, there is always the other side of the coin to be considered.
When something becomes public domain and not just the prerogative of a few fans, the vultures start to come close and turn. And vultures, in this case, are those interested only in monetizing rather than preserving, spreading the voice and sharing.

When more and more people start to take interest in something, it is normal that economic interests also become part of the equation: and so far nothing bad. People have to earn from their work in order to survive or improve their lifestyle. Far from me to criticize those who try to make their passion a job. What I couldn't stand is the arrival of modern conquistadores looking for a new El Dorado. Someone who thinks we're just a fat cow to milk in exchange for peanuts. So, welcome to new projects, new games, new software, new hardware... And I'm willing to pay for it, because other people's work has to be paid, as long as it's done with passion, professionalism and that love that our hobby deserves. I don't want to see things done just to earn money without caring about the passion.

And If most people are attracted to these old computers and want to have more, know more, play more, it is right and proper to provide reliable information and adequate means.
Probably for many of them it is only a momentary interest, the result of the boredom caused by the pandemic closures, and precisely because of this, if we want them to stay, they must be able to count on quality, fairness and passion.
Exactly the things we've always coveted to be there in RetroComputing!

Francesco Fiorentini

Disclaimer

RetroMagazine World as an aperiodic magazine entirely ad-free is a non-profit project and falls off any commercial circuit. All the published material is produced by the respective authors and published thanks to their authorization.

RetroMagazine World is licensed under the terms of: **Attribution-NonCommercial-ShareAlike 4.0 International** (CC BY-NC-SA 4.0) https://creativecommons.org/licenses/by-nc-sa/4.0/

This is a human-readable summary of (and not a substitute for) the license. You are free to:

Share — copy and redistribute the material in any medium or format

Adapt — remix, transform, and build upon the material

The licensor cannot revoke these freedoms as long as you follow the license terms. Under the following terms:

Attribution — You must give appropriate credit, provide a link to the license, and indicate if changes were made. You may do so in any reasonable manner, but not in any way that suggests the licensor endorses you or your use.

NonCommercial — You may not use the material for commercial purposes.

ShareAlike — If you remix, transform, or build upon the material, you must distribute your contributions under the same license as the original.

No additional restrictions — You may not apply legal terms or technological measures that legally restrict others from doing anything the license permits.

RetroMagazine World-English
Year 2 - Issue 7 - MAY 2021

Chief Editor
Francesco Fiorentini
Managing Editor
David La Monaca
Editing Manager
Marco Pistorio
Web Managers
Giorgio Balestrieri/Flavio Soldani